BIG-IP® Access Policy Manager®: Authentication and Single Sign-On

Version 13.1

Table of Contents

LDAP Query..51

RSA SecurID Authentication..57

RADIUS Authentication..73

Authentication Concepts

About AAA server support

Access Policy Manager®(APM®) interacts with authentication, authorization, and accounting (AAA) servers that contain user information. APM supports these AAA servers: RADIUS (authentication and accounting), Active Directory (authentication and query), LDAP (authentication and query), CRLDP, OCSP Responder, TACACS+ (authentication and accounting), SecurID, Kerberos, and HTTP.

A typical configuration includes:

- An APM AAA server configuration object that specifies information about the external AAA server.
- An access policy that includes a logon item to obtain credentials and an authentication item that uses the credentials to authenticate against a specific AAA server.

About AAA high availability support

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established. APM supports these AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.

A typical configuration includes:

- An APM AAA server configuration object that specifies a pool of external AAA servers.
- An access policy that includes a logon item to obtain credentials and an authentication item that uses the credentials to authenticate against one of the servers in the pool.

About AAA and load balancing

When an AAA server supports high availability, you can configure a pool for it in the AAA configuration itself. An AAA server does not load balance over a pool that is attached to a virtual server.

About AAA traffic and route domains

To use route domains for AAA authentication traffic, you must use the pool option in the AAA server configuration. When **Use Pool** is the selected **Server Connection** option, the server address field can take an IP address with route domain (`IPAddress%`*`RouteDomain`*) format. The route domain value is ignored when the AAA server is configured to connect directly to a single server.

About APM support for multiple authentication types

You can add multiple authentication types to an access policy. For example, a user who fails Active Directory authentication might then attempt RADIUS authentication. Or, you might require authentication using a client certificate and then an AAA server.

You can add an authentication item anywhere in the access policy. Typically, you place authentication items somewhere after a logon item.

About APM certificate authentication support

Access Policy Manager® (APM®) supports these types of certificate authentication.

SSL handshake verification and certificate revocation status

APM supports verifying the SSL handshake that occurs at the start of a session or renegotiating the SSL handshake and checking it on demand. A typical configuration includes:

- An access policy that includes a certificate-related access policy item, either Client Cert Inspection or On-Demand Cert Auth.
- A client SSL profile configured per the requirements of Client Cert Inspection or On-Demand Cert Auth.

Note: *If the client SSL profile specifies a certificate revocation list, the access policy item verifies against it.*

Certificate revocation status with OCSP or CRLDP

APM also supports verifying client certificate revocation status with an Online Certificate Status Protocol (OCSP) AAA server or with a Certificate Revocation List Distribution Point (CRLDP) AAA server. A typical configuration includes:

- An AAA server configured to point to an external server (OCSP Responder or CRLDP).
- An access policy that includes either a Client Cert Inspection or an On-Demand Cert Auth access policy item and the appropriate authentication item (OCSP Auth or CRLDP Auth).
- A client SSL profile configured per the requirements of Client Cert Inspection or an On-Demand Cert Auth.

About SSL certificates on the BIG-IP system

Before systems on a network can authenticate one another using SSL, you must install one or more SSL certificates on the BIG-IP® system. An *SSL certificate* is a certificate that a BIG-IP system device presents to another device on the network, for authentication purposes. An SSL certificate can be either a self-signed certificate or a trusted CA certificate.

When you install BIG-IP® software, the application includes a self-signed SSL certificate named `Default`. A *self-signed certificate* is an authentication mechanism that is created and authenticated by the system on which it resides.

If your network includes one or more certificate authority (CA) servers, you can replace the self-signed certificate on each BIG-IP system with a *trusted CA certificate*, that is, a certificate that is signed by a third party. Authenticating BIG-IP systems using trusted CA certificates is more secure than using self-signed certificates.

To ease the task of creating certificate requests and sending them to certificate authorities for signature, the BIG-IP system provides a set of certificate management screens within the BIG-IP Configuration utility.

About local user database support

Access Policy Manager® (APM®) supports authentication against a database that you create on the BIG-IP® system using the Configuration utility. You can employ a local user database for on-box authentication or to control access to external AAA servers.

A typical configuration includes:

- A local user database that you create and populate using the Configuration utility.
- An access policy that includes a local user database authentication item.

About guest access (one-time password) support

Access Policy Manager® (APM®) supports guest access with one-time password generation and verification. A typical configuration includes:

- An SMTP server for sending email or an HTTP AAA server for sending a text message.
- An access policy that includes items to generate a one-time password (OTP), send the generated password to a user, enable the user to log on, and verify the OTP that the user enters.

About authentication for Microsoft Exchange clients

Access Policy Manager® (APM®) supports NTLM and HTTP basic authentication for Microsoft Exchange clients and for this support requires an Exchange profile, created in the Configuration utility. Configuration requirements for NTLM and HTTP basic authentication for Microsoft Exchange clients are otherwise distinct.

Additional resources and documentation for BIG-IP Access Policy Manager

You can access all of the following BIG-IP® system documentation from the AskF5™ Knowledge Base located at `http://support.f5.com/`.

Document	Description
BIG-IP® Access Policy Manager®: Application Access	This guide contains information for an administrator to configure application tunnels for secure, application-level TCP/IP connections from the client to the network.
BIG-IP® Access Policy Manager®: Authentication and Single-Sign On	This guide contains information to help an administrator configure APM for single sign-on and for various types of authentication, such as AAA server, SAML, certificate inspection, local user database, and so on.
BIG-IP® Access Policy Manager®: Customization	This guide provides information about using the APM customization tool to provide users with a personalized experience for access policy screens, and errors. An administrator can apply your organization's brand images and colors, change messages and errors for local languages, and change the layout of user pages and screens.

Document	Description
BIG-IP® Access Policy Manager®: Edge Client and Application Configuration	This guide contains information for an administrator to configure the BIG-IP® system for browser-based access with the web client as well as for access using BIG-IP Edge Client® and BIG-IP Edge Apps. It also includes information about how to configure or obtain client packages and install them for BIG-IP Edge Client for Windows, Mac, and Linux, and Edge Client command-line interface for Linux.
BIG-IP® Access Policy Manager®: Implementations	This guide contains implementations for synchronizing access policies across BIG-IP systems, hosting content on a BIG-IP system, maintaining OPSWAT libraries, configuring dynamic ACLs, web access management, and configuring an access policy for routing.
BIG-IP® Access Policy Manager®: Network Access	This guide contains information for an administrator to configure APM Network Access to provide secure access to corporate applications and data using a standard web browser.
BIG-IP® Access Policy Manager®: Portal Access	This guide contains information about how to configure APM Portal Access. In Portal Access, APM communicates with back-end servers, rewrites links in application web pages, and directs additional requests from clients back to APM.
BIG-IP® Access Policy Manager®: Secure Web Gateway	This guide contains information to help an administrator configure Secure Web Gateway (SWG) explicit or transparent forward proxy and apply URL categorization and filtering to Internet traffic from your enterprise.
BIG-IP® Access Policy Manager®: Third-Party Integration	This guide contains information about integrating third-party products with Access Policy Manager (APM®). It includes implementations for integration with VMware Horizon View, Oracle Access Manager, Citrix Web Interface site, and so on.
BIG-IP® Access Policy Manager®: Visual Policy Editor	This guide contains information about how to use the visual policy editor to configure access policies.
Release notes	Release notes contain information about the current software release, including a list of associated documentation, a summary of new features, enhancements, fixes, known issues, and available workarounds.
Solutions and Tech Notes	Solutions are responses and resolutions to known issues. Tech Notes provide additional configuration instructions and how-to information.

Active Directory Authentication

About Active Directory authentication

You can authenticate using Active Directory authentication with Access Policy Manager®. We support using Kerberos-based authentication through Active Directory.

About Active Directory password management

Access Policy Manager® (APM®) supports password management for Active Directory authentication.

How APM supports password reset

The process works in this sequence:

- Access Policy Manager uses the client's user name and password to authenticate against the Active Directory server on behalf of the client.
- If the user password on the Active Directory server has expired, Access Policy Manager returns a new logon screen back to the user, requesting that the user change the password.
- After the user submits the new password, Access Policy Manager attempts to change the password on the Active Directory server. If this is successful, the user's authentication is validated.

If the password change fails, it is likely that the Active Directory server rejected it because the password did not meet the minimum requirements such as password length.

Number of attempts APM provides for password reset

In the AD Auth action, APM provides a **Max Password Reset Attempts Allowed** property.

Change password option

In the Logon page action, APM provides a Checkbox property in the visual policy editor. You can add the option on the APM logon screen to change the log on password.

About AAA high availability

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established, as usual.

***Note:** Although new authentications fail if the BIG-IP® system loses connectivity to the server, existing sessions are unaffected provided that they do not attempt to re-authenticate.*

APM supports the following AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+. APM supports high availability by providing the option to create a pool of server connections when you configure the supported type of AAA server.

***Note:** If you use AAA with pools, such as RADIUS pools or Active Directory pools, APM assigns each pool member with a different number for the pool member's priority group value. APM must define each* *pool member with a different priority group because AAA load balancing is not used. The priority group number increases automatically with each created pool member. Alternative AAA pool configurations can be defined manually using the full flexibility of Local Traffic Manager™ (LTM®) if load balancing is desired.*

About how APM handles binary values in Active Directory attributes

For Active Directory, Access Policy Manager® (APM®) converts an attribute value to hex only if the value contains unprintable characters. If the session variable contains several values, and one or more of those values is unprintable, then APM converts only those particular values to hex.

An attribute with a single unprintable value

```
7ecc84a2.session.ad.last.attr.objectSid 58 /
 0x01050000000000051500000013fe8e97c03cd5b5ad04e2e255040000
```

Attributes with multiple values, both printable and unprintable (binary)

```
7ecc84a2.session.ad.last.attr.memberOf 460 |
CN=printable group,OU=groups,OU=someco,DC=sherwood,DC=labt,DC=fp,DC=somelabnet,DC=com |
0x434e3d756e7072696e7461626c6520c2bdc2a12067726f75702c4f553d67726f7570732c4f553d66352 | /
c44433d73686572776f6f642c44433d6c6162742c44433d66702c44433d66356e65742c44433d636f6d | /
CN=Domain Users,CN=Users,DC=smith,DC=labt,DC=fp,DC=somlabnet,DC=com | /
CN=CERTSVC_DCOM_ACCESS,CN=Users,DC=smith,DC=labt,DC=fp,DC=somelabnet,DC=com | /
CN=Users,CN=Builtin,DC=smith,DC=labt,DC=fp,DC=somelabnet,DC=com |
```

Task summary for Active Directory authentication

This task list includes all steps required to set up this configuration. If you are adding Active Directory authentication to an existing access profile, you do not need to create another access profile, and the access policy might already include a logon page.

Task list

Configuring an Active Directory AAA server
Creating an access profile
Verifying log settings for the access profile
Configuring Active Directory authentication
Creating a virtual server

Configuring an Active Directory AAA server

You configure an Active Directory AAA server in Access Policy Manager® (APM®) to specify domain controllers for APM to use for authenticating users.

1. On the Main tab, click **Access** > **Authentication** > **Active Directory**.
   The Active Directory Servers list screen opens.
2. Click **Create**.
   The New Server properties screen opens.

3. In the **Name** field, type a unique name for the authentication server.
4. In the **Domain Name** field, type the name of the Windows domain.
5. For the **Server Connection** setting, select one of these options:

   Note: *When configuring an Active Directory AAA server that is located in a nondefault route domain, you must select* ***Use Pool*** *and specify the pool containing the Active Directory server.*

   - Select **Use Pool** to set up high availability for the AAA server.
   - Select **Direct** to set up the AAA server for standalone functionality.
6. If you selected **Direct**, type a name in the **Domain Controller** field.
7. If you selected **Use Pool**, configure the pool:
   a) Type a name in the **Domain Controller Pool Name** field.
   b) Specify the **Domain Controllers** in the pool by typing the IP address and host name for each, and clicking the **Add** button.
   c) To monitor the health of the AAA server, you have the option of selecting a health monitor: only the **gateway_icmp** monitor is appropriate in this case; you can select it from the **Server Pool Monitor** list.
8. In the **Admin Name** field, type a case-sensitive name for an administrator who has Active Directory administrative permissions.

   An administrator name and password are required for an AD Query access policy item to succeed when it includes particular options. Credentials are required when a query includes an option to fetch a primary group (or nested groups), to prompt a user to change password, or to perform a complexity check for password reset.
9. In the **Admin Password** field, type the administrator password associated with the Domain Name.
10. In the **Verify Admin Password** field, retype the administrator password associated with the **Domain Name** setting.
11. In the **Group Cache Lifetime** field, type the number of days.

    The default lifetime is 30 days.
12. In the **Password Security Object Cache Lifetime** field, type the number of days.

    The default lifetime is 30 days.
13. From the **Kerberos Preauthentication Encryption Type** list, select an encryption type.

    The default is **None**. If you specify an encryption type, the BIG-IP® system includes Kerberos preauthentication data within the first authentication service request (AS-REQ) packet.
14. In the **Timeout** field, accept the default value or type a number of seconds.

    The timeout specifies the number of seconds to reach the AAA Active Directory server initially. After the connection is made, the timeout for subsequent operations against the AAA Active Directory server is 180 seconds and is not configurable.
15. Click **Finished**.
    The new server displays on the list.

This adds the new Active Directory server to the Active Directory Servers list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.

3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.

5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.

6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

Note: Logging is disabled when the ***Selected*** *list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring Active Directory authentication

Before you configure an access policy to use Active Directory authentication, you must have at least one Active Directory AAA server configured.

You create an access policy like this one to obtain user credentials and use them to authenticate the user against an external Active Directory server before granting access.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **AD Auth** and click **Add Item**.
   A Properties popup screen opens.
8. From the **Server** list, select the AAA Active Directory server to use for authentication, and click **Save**.
9. You can also set these options.

Option	Description
Cross Domain Support	Specifies whether AD cross domain authentication support is enabled for AD Auth agent.
Complexity check for Password Reset	Specifies whether Access Policy Manager® performs a password policy check. ***Note:*** *Enabling this option increases overall authentication traffic significantly because Access Policy Manager must retrieve additional*

Option	Description
	information. Because this option might require administrative privileges, if you enable it you should specify the administrator name and password on the AAA Active Directory server configuration page.
Show Extended Error	When enabled, displays the comprehensive error messages generated by the authentication server to show on the user's Logon page. This setting is intended for use in testing only in a production or debugging environment. If you enable this setting in a live environment, your system might be vulnerable to malicious attacks
Max Logon Attempts Allowed	Specifies the number of user authentication logon attempts to allow. ***Note:*** *To use this access policy for Citrix Receiver client access, set the value to 1.*
Max Password Reset Attempts Allowed	Specifies the number of times that Access Policy Manager® allows the user to try to change password.

10. Click **Apply Access Policy** to save your configuration.

This adds a logon page and Active Directory authentication to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Creating a virtual server

When creating a virtual server for an access policy, specify an IP address for a single host as the destination address.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **HTTP Profile** list, select **http**.
7. If you use server SSL for this connection, from the **SSL Profile (Server)** list, select a server SSL profile.
8. If you use client SSL for this profile, from the **SSL Profile (Client)** list, select a client SSL profile.
9. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
10. From the **Connectivity Profile** list, select a connectivity profile.

    You can select the default connectivity profile, **connectivity** if you have not defined a specific profile for the traffic that is directed to this virtual server.
11. Click **Finished**.

You have configured a host virtual server and associated an access profile with it.

Testing AAA high availability for supported authentication servers

To effectively test that high availability works for your authentication servers, you should have two servers that are accessible, where you can remove one of them from the network.

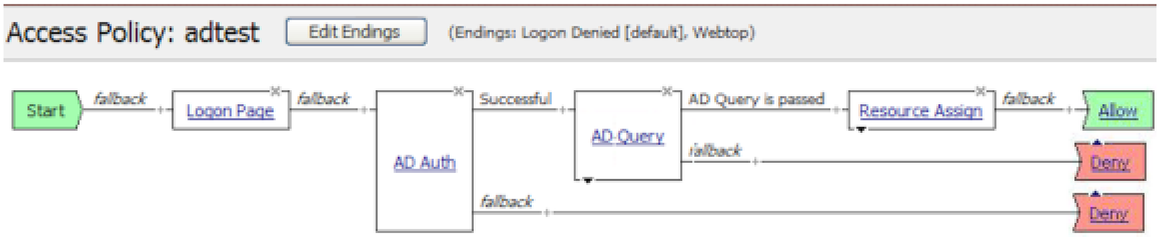
***Note:** High availability is supported for these authentication server types only: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.*

If you configured a supported authentication server type to use a pool of connection servers, you can test the configuration using these steps.

1. Begin a `tcpdump` on the Access Policy Manager®, using a protocol analyzer, and scanning for packets destined for the specific port for your authentication server.
2. Log in to the virtual server with both servers active.
3. Using the `tcpdump` records, verify that the requests are being sent to the higher priority server.
4. Log out of the virtual server.
5. Disable the higher-priority server.
6. Log in to the virtual server again.
7. Verify that the request is being sent to the other server.
8. Log out again, re-enabling the server, and try one more time to verify that the new requests are being sent to the high priority server.

Example access policy using Active Directory authentication and query

This is an example of an access policy with all the associated elements that are needed to authenticate and authorize your users with Active Directory authentication and Active Directory query.



Access Policy: adtest

Figure 1: Example of an access policy for AD auth and query

Importing Active Directory user groups

Import user groups from an Active Directory server to make them available for assigning resources to an Active Directory group. When you configure the AD Group Resource Assign access policy item, you can type group names to exactly match those on the Active Directory server, or you can select them from the imported list of groups.

1. Select **Access** > **Authentication** > **Active Directory**.
   The Active Directory Servers screen displays.
2. Click the name of the server that you want to update.
   The Properties screen displays.

3. From the menu bar, click **Groups**.
4. From the Groups area of the screen, click **Update**.
   The screen displays the number of groups, the date last updated, and the list of groups.

Assigning resources to an AD group

You can select groups from a list that you upload from an Active Directory server; alternately, or in addition. you can type group names to exactly match Active Directory groups. If you plan to select groups and have not updated the list recently, update it from the Groups screen for the AAA Active Directory server before you start.

Use an AD Group Resource Assign action to assign resources to one or more groups that are configured on the Active Directory server. For every group to which a user belongs, the corresponding resources will be assigned to the session.

1. On a policy branch, click the **(+)** icon to add an item to the policy.
   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
2. On the Assignment tab, select the **AD Group Resource Assign** agent, and click **Add Item**.
   The AD Group Resource Assign screen opens, displaying a blank entry in the Groups area.
3. To make a list of groups available, select a server from the **Server** list.

   A brief pause occurs while the agent retrieves any groups that were previously uploaded from the Active Directory server to the BIG-IP® system.
4. To add an entry, click **Add entry**.

   An entry must include at least one group and the resources to be assigned to it. You can add multiple entries.

   A numbered entry displays in the Groups area.
5. In the Groups area, click the **edit** link for the entry that you want to update.
   A popup screen opens to the Groups tab.
6. If you need to add a group, in the **New Group** field, type the name of a group that exists on the server and click **Add group manually**.

   When the access policy runs, this action queries the group names using the **memberOf** attribute in the directory.

   The group displays in the list on the Groups tab.
7. Select at least one group.
8. Repeat these steps for each type of resource that you require.

   The screen displays one tab for each resource type.

   a) Click a tab.
   b) Select the resources that you want to assign to the selected groups.

   Typical resource assignment rules apply. For example, you can assign multiple webtop links to a group, but you can assign only one webtop.
9. Click the **Update** button.
   The **LDAP Group Resource Assign** screen opens, and displays the groups and resources in the entry in the Groups table.
10. Create any additional entries that you require.
11. Click **Save**.
    The properties screen closes and the policy displays.

This configures an AD group resource assign action and adds it to the access policy.

Active Directory authentication session variables

When the AD Auth access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the Active Directory access policy items and for a logon access policy item.

Session variables for Active Directory authentication

Session Variable	Description
`session.ad.last.actualdomain`	AD Auth agent sets this variable to the actual user domain used for successful Active Directory authentication, whether cross-domain support is enabled or disabled.
`session.ad.last.authresult`	Provides the result of the Active Directory authentication. The available values are: • 0: Failed • 1: Passed
`session.ad.last.errmsg`	Displays the error message for the last login. If `session.ad.last.authresult`is set to 0, then `session.ad.last.errmsg` might be useful for troubleshooting purposes.

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

Active Directory cross-domain support rules

Rules	Explanation
Cross-domain support and **split domain from username** are both enabled.	If you enable **cross domain support**, and enable **split domain username** at the login page, and then the user enters his user name, such as `user@domain.com`, Access Policy Manager® uses the `user@domain.com` as the user principal name to authenticate the user against USERNAME.COM domain.
Cross-domain support is enabled but **split domain from username** is disabled	Access Policy Manager handles the user's input as a simple user name and escape "@" and "\" chars. In other words, Access Policy Manager uses `user\@userdomain.com@DEFAULTREALM.COM` to authenticate the user, where DEFAULTREALM.COM is the domain name that was configured on the AAA AD Server configuration page.
If user does not specify a user's domain	Regardless of whether **split domain from username** option is enabled or disabled, Access Policy Manager uses `user@defaultrealm.com` to authenticate the user.

Active Directory authentication and query troubleshooting tips

You might run into problems with Active Directory authentication and query processes in some instances. Follow these tips to try to resolve any issues you might encounter.

Active Directory auth authentication and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`Domain controller reply did not match expectations. (-1765328237)`	This error occurs when the principal/domain name does not match the domain controller server's database. For example, if the actual domain is `SALES.MYCOMPANY.COM`, and the administrator specifies `STRESS` as the domain, then the `krb5.conf` file displays the following: `default_realm = SALES SALES = { domain controller = (domain controller server) admin = (admin server)` So, when the administrator tries to authenticate with `useraccount@SALES`, the krb5 library notices that the principal name `SALES` differs from the actual one in the server database.

Additional troubleshooting tips for Active Directory authentication

You should	Steps to take
Check that your access policy is attempting to perform authentication	• Refer to the message boxes in your access policy to display information on what the access policy is attempting to do. • Refer to `/var/log/apm` to view authentication attempts by the access policy. ***Note:*** *Make sure that your log level is set to the appropriate level. The default log level is* `notice`*.*
Confirm network connectivity	• Access Access Policy Manager® (APM®) through the command line interface and check your connectivity by pinging the Active Directory server using the host entry in the AAA Server box. • Confirm that the Active Directory port (`88` or `389`) is not blocked between APM, and the Active Directory server.
Check the Active Directory server configuration	• Confirm that the Active Directory server name can be resolved to the correct IP address, and that the reverse name resolution (IP address to name) is also possible. • Confirm that the Active Directory server and the BIG-IP® system have the correct time setting configured. ***Note:*** *Since Active Directory is sensitive to time settings, use NTP to set the correct time on the BIG-IP system.*
Capture a tcpdump	Use the tcpdump utility on the BIG-IP system to record activities between Access Policy Manager® and the authentication server when authentication attempts are made. **1.** Type a command to start the tcpdump utility. For example, type `tcpdump -s0 -i 1.1 -w /var/tmp/ad-test.pcap host`

You should	Steps to take
	`10.10.10.10` where `1.1` is an interface number, `/var/tmp/ad-test.pcap` is the path and filename for an output binary file, and `10.10.10.10` is the IP address for the authentication server. ***Note:** For tcpdump utility syntax, refer to SOL411: Overview of packet tracing with the tcpdump utility on the AskF5™ web site located at `support.f5.com`.* **2.** Run the authentication test. **3.** After authentication fails, stop the tcpdump utility, download the result to a client system, and use an analyzer to troubleshoot. ***Important:** If you decide to escalate the issue to customer support, you must provide a capture of the tcpdump when you encounter authentication issues that you cannot otherwise resolve on your own.*

Overview: Using Active Directory Trusted Domains

Active Directory Trusted Domains option in BIG-IP® Access Policy Manager® (APM®) manages Active Directory AAA trusted domains. For enterprises that are service providers, their customers might have their own enterprise network infrastructure. Using APM, the service provider provides access to their customers' networks. To avoid network traffic collisions between two customer networks, the service provider separates each customer using route domains. A *route domain* is a configuration object that isolates network traffic for a particular application on the network. The service provider uses Active Directory to authenticate their customer users. However, each customer's Active Directory service can contain multiple trusted domains or forests. The service provider can use the Active Directory Trusted Domains option to authenticate users across all trusted domains or forests for a customer.

Configuring an Active Directory Trusted Domain

You must create at least one Active Directory AAA server before you can configure an Active Directory Trusted Domain.

Configure an Active Directory Trusted Domain in Access Policy Manager ®(APM®) to authenticate users in route domains with at least one trusted domain.

1. On the Main tab, click **Access** > **Authentication** > **Active Directory** > **Trusted Domains**.
   The Trusted Domains screen opens.
2. Click **Create**.
   The Create New Active Directory Trusted Domains screen opens.
3. In the **Name** field, type a name for the Active Directory Trusted Domain.
4. In the **Description** field, type a description for the Active Directory Trusted Domain.
5. For the **XXX** setting, in the **Available** list, select the Active Directory AAA server that you want to add to the Trusted Domain, and click << to move the Active Directory AAA server into the **Selected** list.
6. From the **Root** list, select a root domain.

   You use the root domain for an initial authentication request, such as an entry point to an Active Directory forest.
7. Click **OK**.

You have now added an Active Directory Trusted Domain to the Active Directory Trusted Domain list.

You can now add the Active Directory Trusted Domain option to either the AD Auth agent or the AD Query agent in the visual policy editor.

Note: *You can select a trusted domain only if you enable the Cross Domain support option.*

Active Directory Query

About Active Directory queries

When running the AD Query access policy item, Access Policy Manager® (APM®) queries an external Active Directory server for additional information about the user. The AD Query item looks up the attribute memberOf to fetch the groups to which a user belongs and provides an additional option to fetch the primary group.

The AD Query item does not authenticate user credentials. To authenticate users, use another or an additional authentication item in the access policy.

About nested groups in Active Directory queries

A *nested group* is a group that is a member of another group. For example, group1 is a member of group3 and group4. A user, user1, that belongs to group1 and group2 also belongs to group3 and group4 through nesting.

Whether AD Query returnd nested groups in session variables

The AD Query access policy item returns and stores the groups to which a user belongs in the *`memberOf`* session variable.

The contents of the *`memberOf`* session variable differ depending on whether the **Fetch Nested Group** setting is enabled or disabled in AD Query properties:

- Enabled - The *`memberOf`* session variable contains all groups to which the user belongs. As in the example, this includes group1, group2, group3, and group4.
- Disabled - The *`memberOf`* session variable contains groups to which the user belongs directly. Based on the example, this would be group1 and group2.

About Active Directory password management

Access Policy Manager® (APM®) supports password management for Active Directory authentication.

How APM supports password reset

The process works in this sequence:

- Access Policy Manager uses the client's user name and password to authenticate against the Active Directory server on behalf of the client.
- If the user password on the Active Directory server has expired, Access Policy Manager returns a new logon screen back to the user, requesting that the user change the password.
- After the user submits the new password, Access Policy Manager attempts to change the password on the Active Directory server. If this is successful, the user's authentication is validated.

If the password change fails, it is likely that the Active Directory server rejected it because the password did not meet the minimum requirements such as password length.

Number of attempts APM provides for password reset

In the AD Auth action, APM provides a **Max Password Reset Attempts Allowed** property.

Change password option

In the Logon page action, APM provides a Checkbox property in the visual policy editor. You can add the option on the APM logon screen to change the log on password.

About how APM handles binary values in Active Directory attributes

For Active Directory, Access Policy Manager® (APM®) converts an attribute value to hex only if the value contains unprintable characters. If the session variable contains several values, and one or more of those values is unprintable, then APM converts only those particular values to hex.

An attribute with a single unprintable value

```
7ecc84a2.session.ad.last.attr.objectSid 58 /
 0x01050000000000051500000013fe8e97c03cd5b5ad04e2e255040000
```

Attributes with multiple values, both printable and unprintable (binary)

```
7ecc84a2.session.ad.last.attr.memberOf 460 |
CN=printable group,OU=groups,OU=someco,DC=sherwood,DC=labt,DC=fp,DC=somelabnet,DC=com |
0x434e3d756e7072696e7461626c6520c2bdc2a12067726f75702c4f553d67726f7570732c4f553d66352 | /
c44433d73686572776f6f642c44433d6c6162742c44433d66702c44433d66356e65742c44433d636f6d | /
CN=Domain Users,CN=Users,DC=smith,DC=labt,DC=fp,DC=somlabnet,DC=com | /
CN=CERTSVC_DCOM_ACCESS,CN=Users,DC=smith,DC=labt,DC=fp,DC=somelabnet,DC=com | /
CN=Users,CN=Builtin,DC=smith,DC=labt,DC=fp,DC=somelabnet,DC=com |
```

Adding an Active Directory query to an access policy

Before you add an AD query to an access policy, you must have at least one AD AAA server configured. You should also have an access policy that is configured with actions to authenticate the user.

You add an AD query to an access policy to get information about a user; for example, you might want to know whether a user is a member of a group before granting access to particular resources. Access Policy Manager® (APM®) stores the attributes it retrieves in session variables.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Authentication tab, select **AD Query** and click **Add Item**.

A Properties popup screen opens.

5. From the **Server** list, select the Active Directory AAA server to query.
6. You can also set these options.

Option	Description
SearchFilter	Type a search filter. (Otherwise if left empty, the policy uses the default filter, sAMAccountName=%{*session.logon.last.username*}. As a result, the `SearchFilter` parameter is populated with the Subject Alternative Name from the current Active Directory session.)
Fetch Primary Group	Enable this setting to populate the user's primary group in the session variables. This setting is optional.
Cross Domain Support	Specifies whether AD cross domain authentication support is enabled for AD Auth agent. This setting is optional.
Fetch Nested Groups	Enable to populate the *memberOf* session variable with user's membership in nested groups in addition to the groups to which the user belongs directly. ***Important:*** *Access Policy Manager does not query for the primary group and add it to the* `memberOf` *attribute. You must manually look up the attribute* `memberOf` *as well as the primary group.*
Complexity Check for Password Reset	Enable this setting so that APM performs the password policy checks it supports.
Max Password Reset Attempts Allowed	Select the number of times to allow a user to try to reset their password.
Prompt user to change password before expiration	Set (N days) to prompt user to change the password before it expires. The default is **none** (disabled). This setting is optional.

7. Click **Save**.
8. Click **Apply Access Policy** to save your configuration.

This adds an Active Directory query to the access policy.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

Note: Logging is disabled when the ***Selected*** *list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Using AD query with IPv6

When you configure an AD AAA server with an IPv6 address in the Domain Controller setting, an AD query does not work. However, we tested AD query with an IPv6 address using this approach.

1. In the AD server configuration, use the host name of the DC in the Domain Controller setting.

```
apm aaa active-directory /Common/AD-IPv6 {
admin-encrypted-password ".(.5(lEhJfN\\<^FaLGC0Bt8CG0KMfR\\9;coEKdIm=5@32II"
admin-name Administrator
domain enterprise.lab.fp.mynet.com
domain-controller win2008.enterprise.lab.fp.mynet.com
```

   The host name is win2008.enterprise.lab.fp.mynet.com in the example.

2. Update the system's global setting to include a remote host entry for the DC host name that was used in step 1 and map it to an IPv4 address as shown in this example.

```
sys global-settings {
gui-setup disabled
hostname bigip2mgmt.lab.fp.mynet.com
mgmt-dhcp disabled
remote-host {
/Common/abc { addr 165.160.15.20
hostname win2008.enterprise.lab.fp.mynet.com
}
}
}
```

3. Create a pool with the DC IPv6 address as a member as shown in this example.

```
ltm pool /Common/AD-IPv6-Pool {
members {
/Common/fd00:ffff:ffff:fff1:912e:cdfe:c884:2607.any {
address fd00:ffff:ffff:fff1:912e:cdfe:c884:2607
}
}
}
```

4. Create a wildcard TCP virtual server with these settings:
   a) Set the **Destination IP** settings to the IPv4 address that was used in step 2.

      That address is 172.31.54.99 in the example.
   b) For the **Service Port** setting, select *** All ports**.
   c) In the Configuration area, leave the **Protocol** setting at the default, **TCP**.
   d) Scroll down to the **Source Address Translation** setting and select **Auto Map**.
   e) Scroll down to the Resources area and select the pool that you configured previously from the **Default Pool** list.

```
ltm virtual /Common/bigip2.lab.fp.mynet.com-tcp {
destination /Common/172.31.54.99:any
ip-protocol tcp
mask 255.255.255.255
pool /Common/AD-IPv6-Pool
profiles {
/Common/tcp { }
}
source-address-translation automap
translate-port disabled
vlans-disabled
}
```

5. Create another similar virtual server, but for UDP traffic. (Set the Protocol setting in the virtual server configuration to UDP).

```
ltm virtual /Common/bigip2.lab.fp.mynet.com-udp {
destination /Common/172.31.54.99:any
ip-protocol udp
mask 255.255.255.255
pool /Common/AD-IPv6-Pool
profiles {
/Common/udp { }
}
source-address-translation automap
translate-port disabled
vlans-disabled
}
```

Active Directory query session variables

When the AD Query access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the Active Directory access policy items and for a logon access policy item.

Session variables for Active Directory query

Session Variable	Description
`session.ad.last.queryresult`	Provides the result of the Active Directory query. The available values are: • 0: Failed • 1: Passed
`session.ad.last.errmsg`	Displays the error message for the last login. If `session.ad.last.queryresult` is set to 0, then `session.ad.last.errmsg` might be useful for troubleshooting purposes.
`session.ad.last.attr.$attr_name`	*`$attr_name`* is a value that represents the user's attributes received from the Active Directory. Each attribute is converted to separate session variables.
`session.ad.last.attr.primarygroup.$attr_name`	*`primarygroup.$attr_name`* is a value that represents the user's group attributes received from the Active Directory. Each attribute is converted to separate session variables.

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

Active Directory authentication and query troubleshooting tips

You might run into problems with Active Directory authentication and query processes in some instances. Follow these tips to try to resolve any issues you might encounter.

Active Directory auth authentication and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`Domain controller reply did not match expectations. (-1765328237)`	This error occurs when the principal/domain name does not match the domain controller server's database. For example, if the actual domain is `SALES.MYCOMPANY.COM`, and the administrator specifies `STRESS` as the domain, then the `krb5.conf` file displays the following: `default_realm = SALES SALES = { domain controller = (domain controller server) admin = (admin server)` So, when the administrator tries to authenticate with `useraccount@SALES`, the krb5 library notices that the principal name `SALES` differs from the actual one in the server database.

Additional troubleshooting tips for Active Directory authentication

You should	Steps to take
Check that your access policy is attempting to perform authentication	• Refer to the message boxes in your access policy to display information on what the access policy is attempting to do. • Refer to `/var/log/apm` to view authentication attempts by the access policy. ***Note:*** *Make sure that your log level is set to the appropriate level. The default log level is `notice`.*
Confirm network connectivity	• Access Access Policy Manager® (APM®) through the command line interface and check your connectivity by pinging the Active Directory server using the host entry in the AAA Server box. • Confirm that the Active Directory port (`88` or `389`) is not blocked between APM, and the Active Directory server.
Check the Active Directory server configuration	• Confirm that the Active Directory server name can be resolved to the correct IP address, and that the reverse name resolution (IP address to name) is also possible. • Confirm that the Active Directory server and the BIG-IP® system have the correct time setting configured.

You should	Steps to take
	Note: *Since Active Directory is sensitive to time settings, use NTP to set the correct time on the BIG-IP system.*
Capture a tcpdump	Use the tcpdump utility on the BIG-IP system to record activities between Access Policy Manager® and the authentication server when authentication attempts are made. **1.** Type a command to start the tcpdump utility. For example, type `tcpdump -s0 -i 1.1 -w /var/tmp/ad-test.pcap host 10.10.10.10` where `1.1` is an interface number, `/var/tmp/ad-test.pcap` is the path and filename for an output binary file, and `10.10.10.10` is the IP address for the authentication server. ***Note:*** *For tcpdump utility syntax, refer to SOL411: Overview of packet tracing with the tcpdump utility on the AskF5™ web site located at* `support.f5.com`*.* **2.** Run the authentication test. **3.** After authentication fails, stop the tcpdump utility, download the result to a client system, and use an analyzer to troubleshoot. ***Important:*** *If you decide to escalate the issue to customer support, you must provide a capture of the tcpdump when you encounter authentication issues that you cannot otherwise resolve on your own.*

LDAP and LDAPS Authentication

About LDAP and LDAPS authentication

You can use LDAPS in place of LDAP when the authentication messages between the Access Policy Manager® and the LDAP server must be secured with encryption. However, there are instances where you will not need LDAPS and the security it provides. For example, authentication traffic happens on the internal side of Access Policy Manager, and might not be subject to observation by unauthorized users. Another example of when not to use LDAPS is when authentication is used on separate VLANs to ensure that the traffic cannot be observed by unauthorized users.

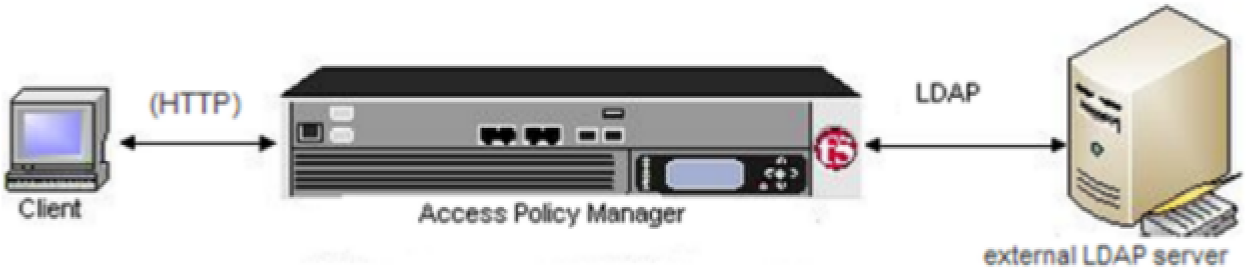


Figure 2: How LDAP works

LDAPS is achieved by directing LDAP traffic over a virtual server that uses server side SSL to communicate with the LDAP server. Essentially, the system creates an LDAP AAA object that has the address of the virtual server. That virtual server (with server SSL) directs its traffic to a pool, which has as a member that has the address of the LDAP server.

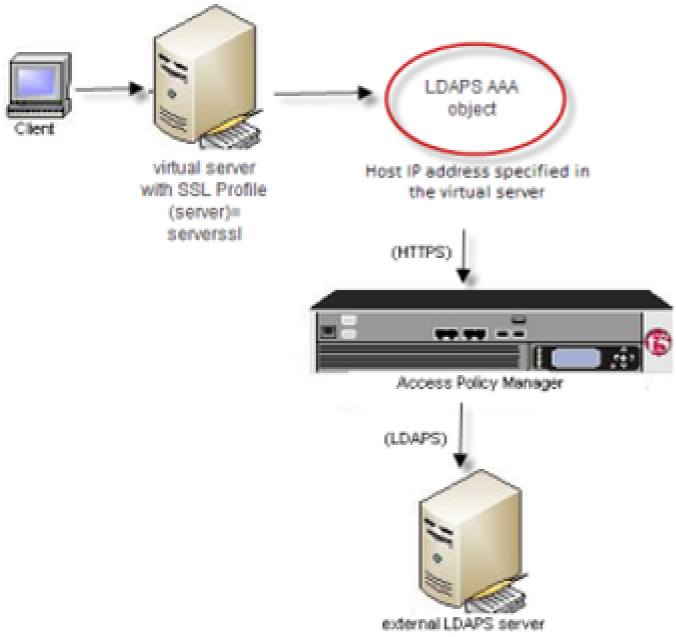


Figure 3: How LDAPS works

About how APM handles binary values in LDAP attributes

For LDAP, Access Policy Manager® (APM®) converts an attribute value to hex only if the value contains unprintable characters. If the session variable contains several values, and one or more of those values is unprintable, then APM converts only those particular values to hex.

An attribute with a single unprintable value

```
9302eb80.session.ldap.last.attr.objectGUID 34 /
0xfef232d3039be9409a72bfc60bf2a6d0
```

Attribute with multiple values, both printable and unprintable (binary)

```
29302eb80.session.ldap.last.attr.memberOf 251 | /
CN=printable group,OU=groups,OU=someco,DC=smith, /
DC=labt,DC=fp,DC=somelabnet,DC=com | /
0x434e3d756e7072696e7461626c6520c2bdc2a12067726f75702c4f553d67726f7570732c4f553d66352c /
44433d73686572776f6f642c44433d6c6162742c44433d66702c44433d66356e65742c44433d636f6d |
```

About AAA high availability

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established, as usual.

Note: *Although new authentications fail if the BIG-IP® system loses connectivity to the server, existing sessions are unaffected provided that they do not attempt to re-authenticate.*

APM supports the following AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+. APM supports high availability by providing the option to create a pool of server connections when you configure the supported type of AAA server.

Note: *If you use AAA with pools, such as RADIUS pools or Active Directory pools, APM assigns each pool member with a different number for the pool member's priority group value. APM must define each pool member with a different priority group because AAA load balancing is not used. The priority group number increases automatically with each created pool member. Alternative AAA pool configurations can be defined manually using the full flexibility of Local Traffic Manager™ (LTM®) if load balancing is desired.*

Task summary for configuring for LDAPS authentication

This task list includes all steps required to set up this configuration. If you are adding LDAPS authentication to an existing access policy, you do not need to create another access profile and the access policy might already include a logon page.

Task list

Configuring an LDAPS AAA server in APM
Creating an access profile
Verifying log settings for the access profile
Configuring LDAPS authentication
Creating a virtual server for LDAPS
Testing LDAPS authentication

Configuring an LDAPS AAA server in APM

You create an LDAPS AAA server when you need to encrypt authentication messages between Access Policy Manager® (APM®) and the LDAP server.

1. Select **Access** > **Authentication** > **LDAP**.
   The LDAP servers screen displays.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For the **Server Connection** setting, select **Use Pool** even if you have only one LDAP server.
5. In the **Server Pool Name** field, type a name for the AAA server pool.
6. Populate the **Server Addresses** field by typing the IP address of a pool member and clicking **Add**.

   Type the IP address of an external LDAP server. If you have more than one pool member, repeat this step.
7. For the **Mode** setting, select `LDAPS`.
8. In the **Service Port** field, retain the default port number for LDAPS, `636`, or type the port number for the SSL service on the server.
9. In the **Admin DN** field, type the distinguished name (DN) of the user with administrator rights.

   Type the value in this format: `CN=administrator,CN=users,DC=sales,DC=mycompany,DC=com`.
10. In the **Admin Password** field, type the administrative password for the server.
11. In the **Verify Admin Password** field, re-type the administrative password for the server.
12. From the **SSL Profile (Server)** list, select an SSL server profile.

    You can select the default profile, serverssl, if you do not need a custom SSL profile.

    LDAPS is achieved by directing LDAP traffic over a virtual server that uses server-side SSL to communicate with the LDAP server.
13. In the **Timeout** field, accept the default value or type a number of seconds.

    The timeout specifies the number of seconds to reach the AAA LDAP server initially. After the connection is made, the timeout for subsequent operations against the AAA LDAP server is 180 seconds and is not configurable.
14. Click **Finished**.
    The new server displays on the list.

The new LDAPS server displays on the LDAP Server list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.

2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.

The access profile log settings display.

4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   *Note: Logging is disabled when the **Selected** list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring LDAPS authentication

You configure an access policy with an LDAP Auth action to provide LDAP authentication for users.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **LDAP Auth** and click **Add Item**.
8. From the **Server** list, select an AAA LDAP server.

   The LDAP Auth action uses SSL connections if you select an LDAP AAA server that is configured for LDAPS.
9. Specify the **SearchDN**, and **SearchFilter** settings.

   SearchDN is the base DN from which the search is done.
10. Click **Save**.
    The properties screen closes and the policy displays.
11. Click **Apply Access Policy** to save your configuration.

This creates a basic access policy that collects credentials and uses them to authenticate with an LDAP server over SSL. In practice, an access policy might include additional types of authentication and might also assign ACLS and resources

***Important:** If you use LDAP Query, Access Policy Manager® does not query for the primary group and add it to the `memberOf` attribute. You must manually look up the attribute `memberOf` as well as the primary group.*

Creating a virtual server for LDAPS

You should have an Access Policy Manager® LDAP AAA server configured in LDAPS mode.

You create a virtual server to handle LDAP traffic and to encrypt authentication messages between Access Policy Manager® and the LDAP server.

***Note:** An AAA server does not load-balance. Do not select a local traffic pool for this virtual server.*

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Configuration** list, select **Advanced**.
5. In the **Destination Address** field, type the IP address for the external LDAP server.

   When you type the IP address for a single host, it is not necessary to append a prefix to the address.

   ***Note:** This IP address must match a server address configured in the LDAP AAA server.*
6. In the **Service Port** field, type the port number for the LDAP server.

   The server port (389) is the virtual port used as the external LDAP server's service port.

   ***Note:** The LDAP AAA server uses the external LDAP server's SSL service port.*
7. From the **SSL Profile (Server)** list, select `serverssl`.

   This ensures the SSL connection between the virtual server and the external LDAP server is in place.
8. From the **Source Address Translation** list, select **Auto Map**.
9. Click **Finished**.

Testing LDAPS authentication

Before starting this procedure, make sure that all the appropriate steps were performed to create an LDAPS authentication.

1. Ensure that LDAP authentication works in your environment.

   An intermediate virtual server should not exist for this verification step.
2. Create an access policy that uses a AAA object that points directly to the LDAP server.
3. Add an intermediate virtual server without a server-side SSL profile.

   Using the same access policy that you just created, modify the AAA object to point to a virtual server.
4. Implement LDAPS by enabling server side SSL, and change the pool member to use port `636`.
5. Review the log messages in Access Policy Manager® reports.
6. Make sure to set the Access Policy log level to **Debug**.

   To set log levels, see **System > Logs > Configurations > Options** > **.**
7. Review the log for LDAP messages and locate and confirm that the bind and search operation succeeds.

Testing AAA high availability for supported authentication servers

To effectively test that high availability works for your authentication servers, you should have two servers that are accessible, where you can remove one of them from the network.

***Note:** High availability is supported for these authentication server types only: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.*

If you configured a supported authentication server type to use a pool of connection servers, you can test the configuration using these steps.

1. Begin a `tcpdump` on the Access Policy Manager®, using a protocol analyzer, and scanning for packets destined for the specific port for your authentication server.
2. Log in to the virtual server with both servers active.
3. Using the `tcpdump` records, verify that the requests are being sent to the higher priority server.
4. Log out of the virtual server.
5. Disable the higher-priority server.
6. Log in to the virtual server again.
7. Verify that the request is being sent to the other server.
8. Log out again, re-enabling the server, and try one more time to verify that the new requests are being sent to the high priority server.

Example of LDAP auth and query default rules

In this example, after successful authentication, the system retrieves a user group using an LDAP query. Resources are assigned to users and users are directed to a webtop if the user group has access to the network access resources.

In this figure, the default branch rule for LDAP query was changed to check for a specific user group attribute.

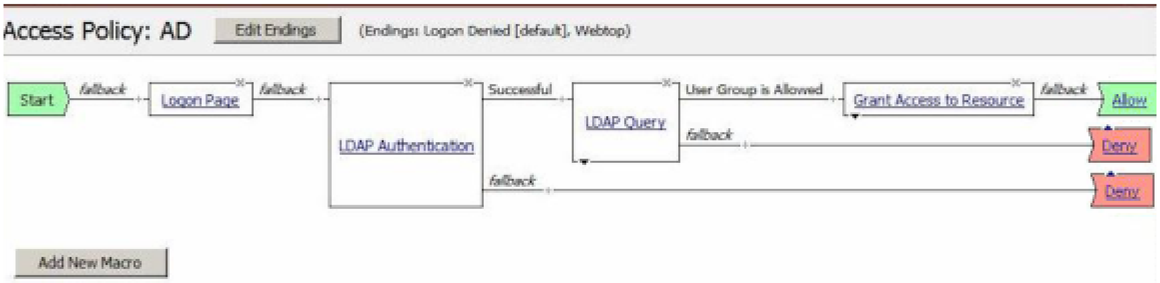


Figure 4: Example of an access policy for LDAP auth query

Importing LDAP user groups

Import user groups from an LDAP server to make them available for assigning resources to an LDAP group. When you configure the LDAP Group Resource Assign access policy item, you can type group names to exactly match those on the LDAP server, or you can select them from the imported list of groups.

1. Select **Access** > **Authentication** > **LDAP**.

The LDAP servers screen displays.

2. Click the name of the server that you want to update.
   The Properties screen displays.
3. From the menu bar, click **Groups**.
4. From the Groups area of the screen, click **Update**.
   After uploading the list, the screen displays the number of groups, the date last updated, and the list of groups.

Assigning resources to an LDAP group

You can select groups from a list that you upload from an LDAP server; alternately, or in addition. you can type group names to exactly match LDAP groups. If you plan to select groups and have not updated the list recently, update it from the Groups screen for the AAA LDAP server before you start.

Use an LDAP Group Resource Assign action to assign resources to one or more groups that are configured on the LDAP server. For every group to which a user belongs, the corresponding resources will be assigned to the session.

1. On a policy branch, click the **(+)** icon to add an item to the policy.
   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
2. On the Assignment tab, select the **LDAP Group Resource Assign** agent, and click **Add Item**.
   The LDAP Group Resource Assign screen opens.
3. To make a list of groups available, select a server from the **Server** list.

   A brief pause occurs while the agent retrieves any groups that were previously uploaded from the LDAP server to the BIG-IP® system.
4. To add an entry, click **Add entry**.

   An entry must include at least one group and the resources to be assigned to it. You can add multiple entries.

   A numbered entry displays in the Groups area.
5. In the Groups area, click the **edit** link for the entry that you want to update.
   A popup screen opens to the Groups tab.
6. If you need to add a group, in the **New Group** field, type the name of a group that exists on the server and click **Add group manually**.

   When the access policy runs, this action queries the group names using the **memberOf** attribute in the directory.

   The group displays in the list on the Groups tab.
7. Select at least one group.
8. Repeat these steps for each type of resource that you require.

   The screen displays one tab for each resource type.

   a) Click a tab.
   b) Select the resources that you want to assign to the selected groups.

   Typical resource assignment rules apply. For example, you can assign multiple webtop links to a group, but you can assign only one webtop.
9. Click the **Update** button.
   The **LDAP Group Resource Assign** screen opens, and displays the groups and resources in the entry in the Groups table.
10. Create any additional entries that you require.
11. Click **Save**.
    The properties screen closes and the policy displays.

This configures an LDAP group resource assign action and adds it to the access policy.

LDAP authentication session variables

When the LDAP Auth access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the LDAP Auth access policy items and for a logon access policy item.

Session variables for LDAP authentication

Session Variable	Description
`session.ldap.last.authresult`	Provides the result of the LDAP authentication. The available values are: • 0: Failed • 1: Passed
`session.ldap.last.errmsg`	Useful for troubleshooting, and contains the last error message generated for LDAP, for example `aad2a221.ldap.last.errmsg`.

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

UserDN settings in LDAP

The following is an example of a typical UserDN usage for LDAP.

Access Policy Manager® attempts to bind with the LDAP server using the supplied DN and user-entered password. If the bind succeeds, that is, authentication succeeds, the user is validated. If the bind fails, the authentication fails. This value is a fully qualified DN of the user with rights to run the query. Specify this value in lowercase and without spaces to ensure compatibility with some specific LDAP servers. The specific content of this string depends on your directory layout.

For example, in an LDAP structure, a typical UserDN for query would be similar to the following string: *`cn=%{session.logon.last.username}, cn=users, dc=sales, dc=com.`*

Access Policy Manager supports using session variables in the **SearchFilter**, **SearchDN**, and **UserDN** settings.

For example, if you want to use the user's CN from the user's SSL certificate as input in one of these fields, you can use the session variable *`session.ssl.cert.last.cn`* in place of *`session.logon.last.username`*.

LDAP authentication and query troubleshooting tips

You might run into problems with LDAP authentication and query in some instances. Follow these tips to try to resolve any issues you might encounter.

LDAP auth and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`LDAP auth failed`	• User name or password does not match records. • No LDAP server is associated with the LDAP Auth agent. • The target LDAP server host/port information associated with the LDAP Auth agent might be invalid. • The target LDAP service might be not accessible.
`LDAP query failed`	• The specified administrative credential is incorrect. • If no administrative credential is specified, then the user name or password does not match. • No LDAP server is associated with the LDAP query agent. • The target LDAP server host/port information associated with the LDAP query agent might be invalid. • The target LDAP service might be not accessible. • If the LDAP query is successfully, then check whether the LDAP query Rules are properly configured.

Additional troubleshooting tips for LDAP authentication

You should	Steps to take
Check that your access policy is attempting to perform authentication	• Refer to the message boxes in your access policy to display information on what the access policy is attempting to do. • Refer to`/var/log/apm` to view authentication attempts by the access policy. ***Note:** Make sure that your log level is set to the appropriate level. The default log level is* `notice`
Confirm network connectivity	• Access the Access Policy Manager® through the command line interface and check your connectivity by pinging the LDAP server using the host entry in the AAA Server box. • Confirm that the LDAP port `389` is not blocked between the Access Policy Manager and the LDAP server.
Confirm network connectivity	• Access the Access Policy Manager through the command line interface and check your connectivity by pinging the LDAP server using the host entry in the AAA Server box. • Confirm that the LDAP port `389` is not blocked between the Access Policy Manager and the LDAP server.
Check the LDAP server configuration	• Verify that the administrative credentials are correct on the LDAP server, and that they match the credentials used by the AAA entry.

You should	Steps to take
	Note: *A good test is to use full administrative credentials with all rights. If that works, you can use less powerful credentials for verification.*
Capture a tcpdump	Use the tcpdump utility on the BIG-IP system to record activities between Access Policy Manager® and the authentication server when authentication attempts are made. **1.** Type a command to start the tcpdump utility. For example, type `tcpdump -s0 -i 1.1 -w /var/tmp/ldap-test.pcap host 10.10.10.10` where `1.1` is an interface number, `/var/tmp/ldap-test.pcap` is the path and filename for the output binary file, and `10.10.10.10` is the IP address for the authentication server. ***Note:*** *For tcpdump utility syntax, refer to SOL411: Overview of packet tracing with the tcpdump utility on the AskF5™ web site located at* `support.f5.com`*.* **2.** Run the authentication test. **3.** After authentication fails, stop the tcpdump utility, download the result to a client system, and use an analyzer to troubleshoot. ***Important:*** *If you decide to escalate the issue to customer support, you must provide a capture of the tcpdump when you encounter authentication issues that you cannot otherwise resolve on your own.*

LDAP Query

About LDAP queries

When running the LDAP Query access policy item, Access Policy Manager® (APM®) queries an external LDAP server for additional information about the user.

Important: *If you use LDAP query, Access Policy Manager does not query for the primary group and add it to the* `memberOf` *attribute. You must look up the attribute* `memberOf`*, as well as the primary group, manually.*

The LDAP Query item does not authenticate user credentials. To authenticate users, use another or an additional authentication item in the access policy.

About how APM handles binary values in LDAP attributes

For LDAP, Access Policy Manager® (APM®) converts an attribute value to hex only if the value contains unprintable characters. If the session variable contains several values, and one or more of those values is unprintable, then APM converts only those particular values to hex.

An attribute with a single unprintable value

```
9302eb80.session.ldap.last.attr.objectGUID 34 /
0xfef232d3039be9409a72bfc60bf2a6d0
```

Attribute with multiple values, both printable and unprintable (binary)

```
29302eb80.session.ldap.last.attr.memberOf 251 | /
CN=printable group,OU=groups,OU=someco,DC=smith, /
DC=labt,DC=fp,DC=somelabnet,DC=com | /
0x434e3d756e7072696e7461626c6520c2bdc2a12067726f75702c4f553d67726f7570732c4f553d66352c /
44433d73686572776f6f642c44433d6c6162742c44433d66702c44433d66356e65742c44433d636f6d |
```

Adding an LDAP query to an access policy

Before you add an LDAP query to an access policy, you must have at least one LDAP AAA server configured. You should also have an access profile that is configured with actions to authenticate the user.

You add an LDAP query to an access policy to get information about a user. Access Policy Manager® (APM®) stores the attributes it retrieves in session variables.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.

3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Authentication tab, select **LDAP Query** and click **Add Item**.
5. From the **Server** list, select an AAA LDAP server.

   An LDAP Query uses SSL connections when you select an LDAP AAA server that is configured for LDAPS.
6. Specify the **SearchDN**, and **SearchFilter** settings.

   SearchDN is the base DN from which the search is done.
7. Click **Save**.
   The properties screen closes and the policy displays.
8. Click **Apply Access Policy** to save your configuration.

This adds an LDAP Query to an existing access policy.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Example of LDAP auth and query default rules

In this example, after successful authentication, the system retrieves a user group using an LDAP query. Resources are assigned to users and users are directed to a webtop if the user group has access to the network access resources.

In this figure, the default branch rule for LDAP query was changed to check for a specific user group attribute.

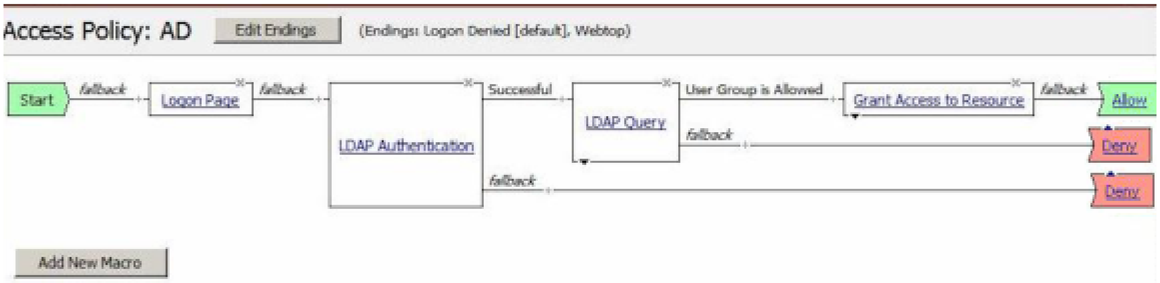


Figure 5: Example of an access policy for LDAP auth query

Session variables in LDAP query properties

You can use session variables to configure properties for the LDAP query access policy item. The properties are listed in the table.

Property	Example value	Description
SearchFilter	`(sAMAccountName=%{session.logon.last.username})`	Populates the `SearchFilter` parameter with the username from the current session.
UserDN	cn=%`{session.logon.last.username}`, cn=users, dc=sales, dc=com.	A typical UserDN for query in an LDAP structure.
SearchDN	`session.ssl.cert.last.cn`	Uses the user CN from the SSL certificate. Useful as a value for any property in this table.

LDAP query session variables

When the LDAP Query access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the LDAP query access policy item and for a logon access policy item.

Session variables for LDAP query

Session Variable	Description
`session.ldap.last.queryresult`	Provides the result of the LDAP query. The available values are: • 0: Failed • 1: Passed
`session.ldap.last.attr.$attr_name`	*`$attr_name`* is a value that represents the user's attributes received during LDAP/query. Each attribute is converted to separate session variables.
`session.ldap.last.errmsg`	Contains only a simple error message for the last error generated for LDAP.

Session Variable	Description
`session.ldap.last.errmsgext`	Useful for troubleshooting. At any log level, contains extended error information for the last error message generated for LDAP.

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

LDAP authentication and query troubleshooting tips

You might run into problems with LDAP authentication and query in some instances. Follow these tips to try to resolve any issues you might encounter.

LDAP auth and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`LDAP auth failed`	• User name or password does not match records. • No LDAP server is associated with the LDAP Auth agent. • The target LDAP server host/port information associated with the LDAP Auth agent might be invalid. • The target LDAP service might be not accessible.
`LDAP query failed`	• The specified administrative credential is incorrect. • If no administrative credential is specified, then the user name or password does not match. • No LDAP server is associated with the LDAP query agent. • The target LDAP server host/port information associated with the LDAP query agent might be invalid. • The target LDAP service might be not accessible. • If the LDAP query is successfully, then check whether the LDAP query Rules are properly configured.

Additional troubleshooting tips for LDAP authentication

You should	Steps to take
Check that your access policy is attempting to perform authentication	• Refer to the message boxes in your access policy to display information on what the access policy is attempting to do. • Refer to`/var/log/apm` to view authentication attempts by the access policy. **Note:** *Make sure that your log level is set to the appropriate level. The default log level is* `notice`

You should	Steps to take
Confirm network connectivity	• Access the Access Policy Manager® through the command line interface and check your connectivity by pinging the LDAP server using the host entry in the AAA Server box. • Confirm that the LDAP port `389` is not blocked between the Access Policy Manager and the LDAP server.
Confirm network connectivity	• Access the Access Policy Manager through the command line interface and check your connectivity by pinging the LDAP server using the host entry in the AAA Server box. • Confirm that the LDAP port `389` is not blocked between the Access Policy Manager and the LDAP server.
Check the LDAP server configuration	• Verify that the administrative credentials are correct on the LDAP server, and that they match the credentials used by the AAA entry. ***Note:*** *A good test is to use full administrative credentials with all rights. If that works, you can use less powerful credentials for verification.*
Capture a tcpdump	Use the tcpdump utility on the BIG-IP system to record activities between Access Policy Manager® and the authentication server when authentication attempts are made. **1.** Type a command to start the tcpdump utility. For example, type `tcpdump -s0 -i 1.1 -w /var/tmp/ldap-test.pcap host 10.10.10.10` where `1.1` is an interface number, `/var/tmp/ldap-test.pcap` is the path and filename for the output binary file, and `10.10.10.10` is the IP address for the authentication server. ***Note:*** *For tcpdump utility syntax, refer to SOL411: Overview of packet tracing with the tcpdump utility on the AskF5™ web site located at `support.f5.com`.* **2.** Run the authentication test. **3.** After authentication fails, stop the tcpdump utility, download the result to a client system, and use an analyzer to troubleshoot. ***Important:*** *If you decide to escalate the issue to customer support, you must provide a capture of the tcpdump when you encounter authentication issues that you cannot otherwise resolve on your own.*

RSA SecurID Authentication

About RSA SecurID authentication

RSA SecurID is a two-factor authentication mechanism based on a one-time passcode (OTP) that is generated by using a token code provided by a software or hardware authenticator. Both BIG-IP® Edge Client® for Windows and OS X systems support the RSA SecurID feature.

A *token* is a one-time authentication code generated every 60 seconds by an authenticator (hardware or software) assigned to the user.

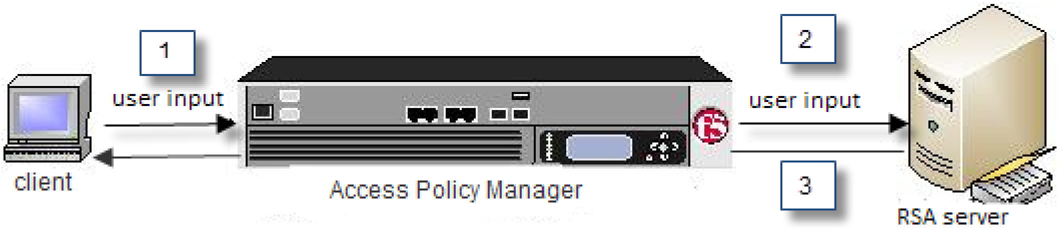


Figure 6: How Access Policy Manager works with RSA SecurID

1. Access Policy Manager® displays the logon page to the user. The logon page accepts the username and PIN code provided by the user, requests the passcode from RSA Secure-ID Software Token software, and sends the username and passcode to Access Policy Manager.
2. Access Policy Manager sends the user-specified inputs to the RSA authentication server.
3. Based on the authentication results, Access Policy Manager grants or denies access to the client.

About SecurID configuration requirements for APM AAA

Before you can use a SecurID AAA server in Access Policy Manager® (APM®), you must configure specific elements and settings on RSA SecurID. To provide RSA SecurID authentication for APM, RSA Authentication Manager requires:

- An authentication agent for APM in its database.
- A RADIUS client that corresponds to the authentication agent for APM.
- A SecurID token policy .

Authentication agent

To create an authentication agent from the RSA Security Console, you need to provide this information:

- Hostname
- IP addresses for all network interfaces

Note: *You must set Agent Type to Standard Agent.*

RADIUS client

To create a RADIUS client from the RSA Security Console, you need to provide this information:

- Hostname
- IP addresses for all network interface

- RADIUS secret (this RADIUS secret must match the corresponding RADIUS secret on the APM system).

Character requirements setting in a SecurID token policy

To avoid a problem in the RSA SDK with alphabetic-only PIN policies, do not use them. When you set up a SecurID token policy, set the character requirements to one of these values:

- Require numeric PINs
- Allow alpha-numeric PINs

About SecurID configuration requirements for high availability

For high availability to work with RSA SecurID, the same node secret is required on both members of the HA pair.

In addition, one of these configurations must be in place for IP addresses.

One alternative is to use a SNAT Pool on the BIG-IP system to force the same source IP address. For more information, see SOL15517: Specifying the source IP address used by BIG-IP APM AAA servers on the AskF5™ web site located at `http://support.f5.com/`.

Instead of using a SNAT Pool on the BIG-IP system, both self IP addresses can be specified in the RSA host agent definition. The definition must include:

- A floating IP address, created from the RSA server, for the host agent.
- The static self IP addresses of the nodes defined as secondary nodes.

The Configuration file generated on the RSA server contains all three IP addresses, so the originating traffic from any of the sub-nodes will be accepted.

Task summary for configuring for RSA SecurID authentication

This task list includes all steps required to set up this configuration and provides an example access policy that uses both RSA SecurID and Active Directory authentication. It is only an example. If you are adding RSA SecurID authentication to an existing access policy, you do not need to create another access profile.

Task list

Configuring a SecurID AAA server in APM

Creating an access profile

Verifying log settings for the access profile

Configuring RSA SecurID authentication in an access policy

Creating a virtual server

Configuring a SecurID AAA server in APM

Configure a SecurID AAA server for Access Policy Manager® (APM®) to request RSA SecurID authentication from an RSA Manager authentication server.

1. On the Main tab, click **Access** > **Authentication**.
   The Authentication screen opens.
2. On the menu bar, click **AAA Servers By Type**, and select **SecurID**.
   The SecurID screen opens and displays the servers list.
3. Click **Create**.

The New Server properties screen opens.

4. In the **Name** field, type a unique name for the authentication server.
5. In the Configuration area, for the **Agent Host IP Address (must match the IP address in SecurID Configuration File)** setting, select an option as appropriate:
   - **Select from Self IP List**: Choose this when there is no NAT device between APM and the RSA Authentication Manager. Select an IP from the list of those configured on the BIG-IP® system (in the Network area of the Configuration utility).
   - **Other**: Choose this when there is a NAT device in the network path between Access Policy Manager and the RSA Authentication Manager server. If selected, type the address as translated by the NAT device.

   ***Note:** This setting does not change the source IP address of the packets that are sent to the RSA SecurID server. (Layer 3 source addresses remain unchanged.) The agent host IP address is used only in Layer 7 (application layer) information that is sent to the RSA SecurID server.*
6. For the **SecurID Configuration File** setting, browse to upload the `sdconf.rec` file.

   Consult your RSA Authentication Manager administrator to generate this file for you.
7. Click **Finished**.
   The new server displays on the list.

This adds a new RSA SecurID server to the AAA Servers list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   ***Note:** A access profile name must be unique among all access profile and any per-request policy names.*
4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     ***Note:** No access policy is associated with this type of access profile*
   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).

- **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

  Note: You can edit Identity Service profile properties.

Note: Depending on licensing, you might not see all of these profile types.

Additional settings display.

5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring RSA SecurID authentication in an access policy

Before you add RSA SecurID authentication to an access policy, you must have at least one AAA SecurID server configured in Access Policy Manager® (APM®). You might need an AAA server configured for another type of authentication, depending on the number of authentication actions that you plan to add to this access policy. This access policy uses Active Directory authentication in addition to SecurID; in this case, an Active Directory AAA server is required.

You add RSA SecurID authentication to an access policy so that APM can request RSA SecurID authentication using the AAA SecurID server that you specify.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.

2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. To customize the Logon Page to prompt for a token code in addition to a password, perform these substeps:

   Add a second password field to the logon page and supply the appropriate prompts for both password fields.

   a) From the **Type** list in row 3, select **password**.
   b) In the **Post Variable Name** field in row 3, type `password1`.

      The name password1 is an example.
   c) From the **Session Variable Name** field in row 3, type `password1`.

      The name password1 is an example. If you type `password1`, the name password1 becomes part of the session variable name, *`session.logon.last.password1`*. APM stores user input for the field in this session variable.

      You now have two fields that accept passwords on this Logon Page. Next you must set the prompts that display for each password field. This access policy runs RSA SecurID authentication first and another type of authentication afterward.
   d) In the Customization area in **Logon Page Input Field #2**, in place of the text `Password` type `RSA Token` or the wording of your choice,
   e) In **Logon Page Input Field #3**, type a prompt for the other type of authentication, for example `Password`.
   f) Click **Save**.
      The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **RSA SecurID** and click **Add Item**.
   A properties popup screen opens.
8. From the **AAA Server** list in the properties popup screen, select the SecurID AAA server that you want to associate to the agent.
9. Set **Max Logon Attempts** to a value from from 1 to 5.

   Note: *To use this access policy for Citrix Receiver client access, you must set **Max Logon Attempts** to 1.*
10. Click **Save**.
    The properties screen closes and the policy displays.
11. Add a Variable Assign action after the Logon Page action.

Authentication actions use the password in the *`session.last.logon.password`* session variable. When the access policy runs and reaches this point, the RSA token code is stored in that session variable.

After you add the Variable Assign action, a Properties popup screen displays.

12. On the Properties screen, add an entry to replace the contents of the *`session.last.logon.password`* session variable with the password stored in the *`session.last.logon.password1`* session variable:

a) Click **Add new entry**.
An **empty** entry appears in the Assignment table.

b) Click the **change** link in the new entry.
A popup screen opens.

c) From the left-side list, select **Custom Variable** (the default), and type `session.logon.last.password`.

d) From the right-side list, select **Custom Expression** (the default), and type `expr { [mcget -secure {session.logon.last.password1}] }`.

e) Click **Finished**.

The popup screen closes.

f) Click **Save**.
The properties screen closes and the policy displays.

This example adds an AD Auth access policy item as a second type of authentication. You can add an authentication access policy item other than AD Auth.

The *`session.logon.last.password`* session variable now contains the user-entered password.

13. On the fallback branch after the previous action, click the **(+)** icon to add an item to the policy.
A popup screen opens.

14. On the Authentication tab, select **AD Auth**.
A properties screen displays.

15. From the **Server** list, select a server.

16. To support Citrix Receiver clients, you must set **Max Logon Attempts** to 1.

17. Click **Save**.
The properties screen closes and the policy displays.

18. Add another authentication action and any other actions you require.

19. Click **Apply Access Policy** to save your configuration.

This adds RSA SecurID AAA authentication to the access policy and a second type of authentication.

Creating a virtual server

When creating a virtual server for an access policy, specify an IP address for a single host as the destination address.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **HTTP Profile** list, select **http**.

7. If you use server SSL for this connection, from the **SSL Profile (Server)** list, select a server SSL profile.
8. If you use client SSL for this profile, from the **SSL Profile (Client)** list, select a client SSL profile.
9. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
10. From the **Connectivity Profile** list, select a connectivity profile.

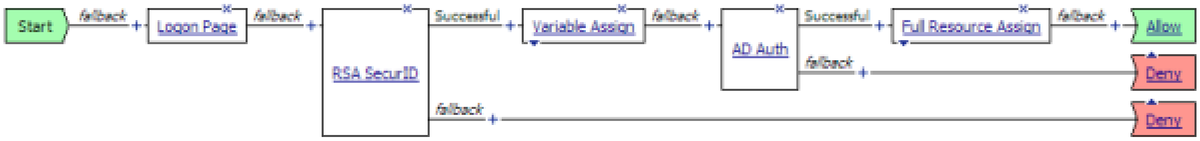
    You can select the default connectivity profile, **connectivity** if you have not defined a specific profile for the traffic that is directed to this virtual server.
11. Click **Finished**.

You have configured a host virtual server and associated an access profile with it.

Access policy example for RSA and AD authentication

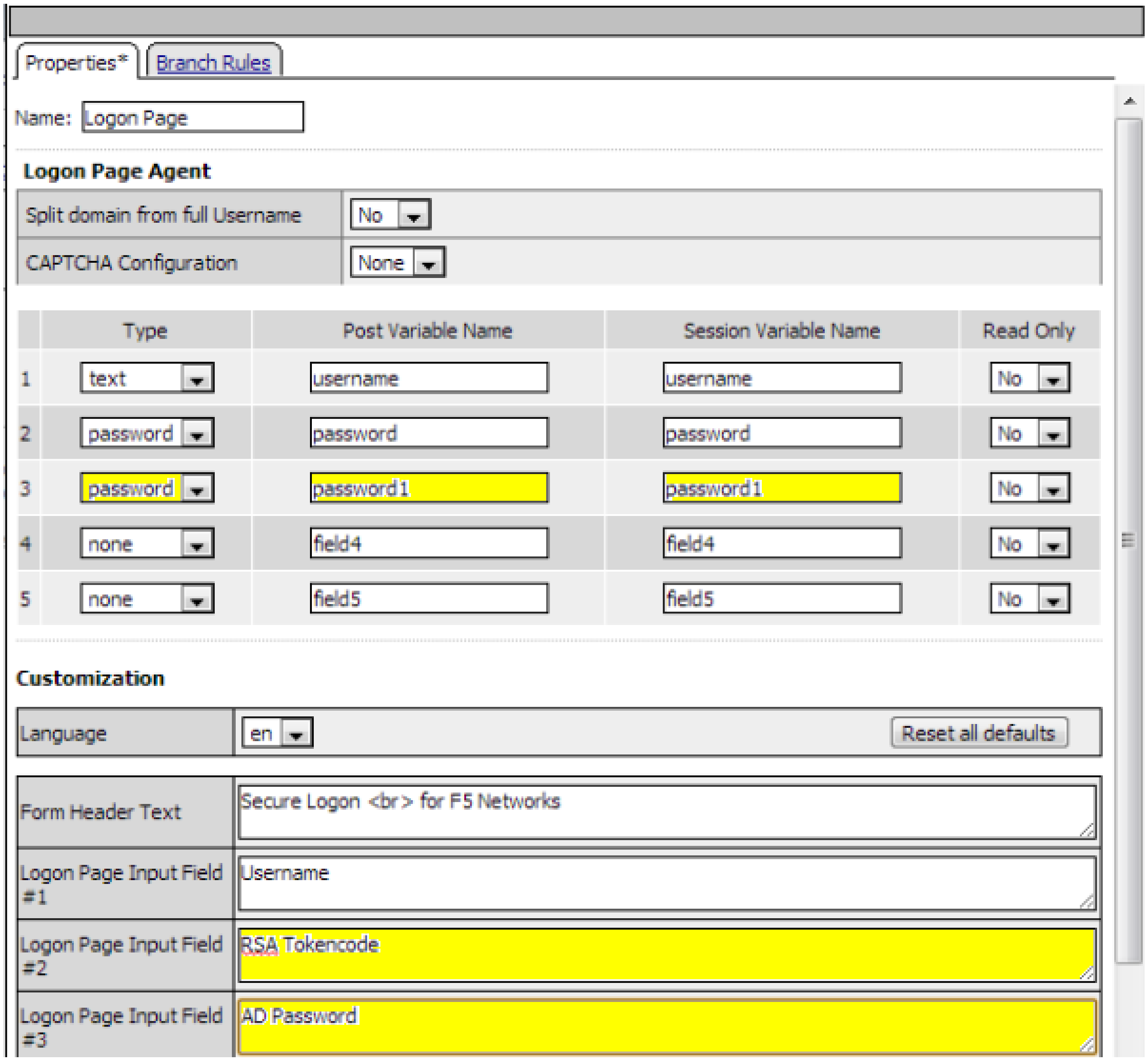
Typically, when you configure an authentication action, you precede it with a Logon Page action to collect credentials. This example describes how to include more than one authentication item (RSA and AD authentication) in an access policy and present a Logon Page only once.

Access policy with RSA SecurID and AD Auth actions



In this example, if the Logon Page action is not customized, the access policy passes the same credentials to both the RSA SecurID and AD Auth authentication agents. But RSA SecurID accepts a user name and a token at logon, while Active Directory accepts a user name and password. To accommodate these differences, customize the Logon Page item.

Logon Page customization: how to collect a token and a password



Name: Logon Page

Logon Page Agent

Split domain from full Username	No
CAPTCHA Configuration	None

	Type	Post Variable Name	Session Variable Name	Read Only
1	text	username	username	No
2	password	password	password	No
3	password	password1	password1	No
4	none	field4	field4	No
5	none	field5	field5	No

Customization

Language	en
Form Header Text	Secure Logon for F5 Networks
Logon Page Input Field #1	Username
Logon Page Input Field #2	RSA Tokencode
Logon Page Input Field #3	AD Password

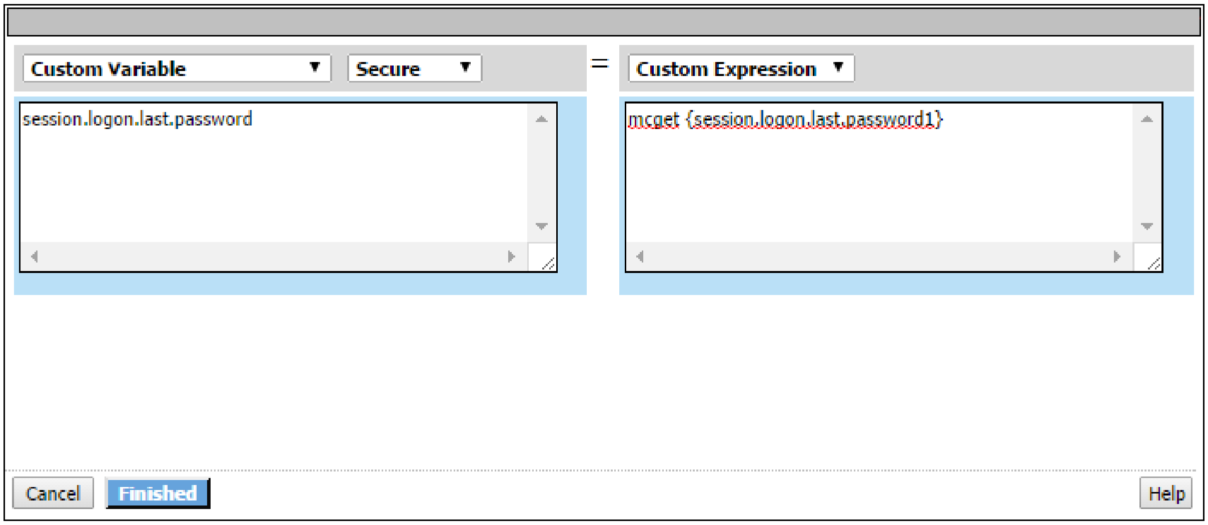
The first highlighted entry defines a second password field. The second password is stored in the `session.variable.last.password1` variable.

***Note:** Although the second password is stored in a session variable, it is not the session variable,* `session.variable.last.password`*, from which an authentication agent accepts the password.*

The highlighted entries in the Customization area change the labels that the Logon Page displays, from Password to `RSA Token Code` for the first password and to `AD Password` for the second password.

Variable Assign action: How to pass the AD Password to the AD Auth

Use the Variable Assign action to provide the appropriate password before the AD Auth action occurs.



```
session.logon.last.password
```

```
mcget {session.logon.last.password1}
```

The Variable Assign action moves the AD Auth password, stored in `session.variable.last.password1`, to the `session.variable.last.password` variable.

RSA SecurID session variables for access policy rules

When the RSA SecurID access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the RSA SecurID access policy item and a logon access policy item.

Session variables for RSA SecurID

Session Variable	Description
`session.securid.last.result`	Provides the result of the RSA SecurID authentication. The available values are: • 0: Failed • 1: Passed

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

RSA SecurID on Windows using RADIUS configuration troubleshooting tips

You might run into problems with RSA SecurID on Windows using RADIUS configuration. Follow these tips to try to resolve any issues that you encounter.

RSA SecurID on Windows using RADIUS configuration troubleshooting

Possible error messages	Possible explanations and corrective actions
`The RADIUS server is inactive`	Even if the RADIUS server was started from the SecurID options window on the Windows SecurID server, the server might not be active. In Windows Services Manager, make sure that the server is set to start each time the server boots, and is currently running. RSA SecurID authentication using RADIUS takes place on a different port than the native securid ID.
`The SecurID is configured incorrectly for RADIUS authentication`	While using RSA SecurID over RADIUS, the SecurID server is a client of itself. The RADIUS service functions as a standalone process, and if the SecurID server is not set up as a client of itself, it rejects the Access Policy Manager ®authentication request and does not store anything in the logs.
`No response from the RSA SecurID server`	Check that RSA Authentication Manager is configured properly. To facilitate communication between Access Policy Manager and the RSA Authentication Manager, you must add an Authentication Agent record to the RSA Authentication Manager database. The Authentication Agent record identifies the Access Policy Manager within its database, and contains information about communication and encryption. To create the Authentication Agent record, you need this information. • Host name • IP addresses for all network interfaces When adding the Authentication Agent record, you should configure the Access Policy Manager as a Standard Agent. The RSA Authentication Manager uses this setting to determine how to communicate with Access Policy Manager. You must also add a RADIUS client that corresponds to the Authentication Agent. To create the RADIUS client, you need this information. • Host name • IP addresses for all network interfaces • RADIUS secret (This RADIUS secret must match the corresponding RADIUS secret on the Access Policy Manager.)

About BIG-IP Edge Client RSA SecurID authentication

RSA SecurID is a two-factor authentication mechanism based on a one-time passcode (OTP) that is generated by using a token code provided by a software or hardware authenticator. Both BIG-IP® Edge Client® for Windows and OS X systems support the RSA SecurID feature.

A *token* is a one-time authentication code generated every 60 seconds by an authenticator (hardware or software) assigned to the user.

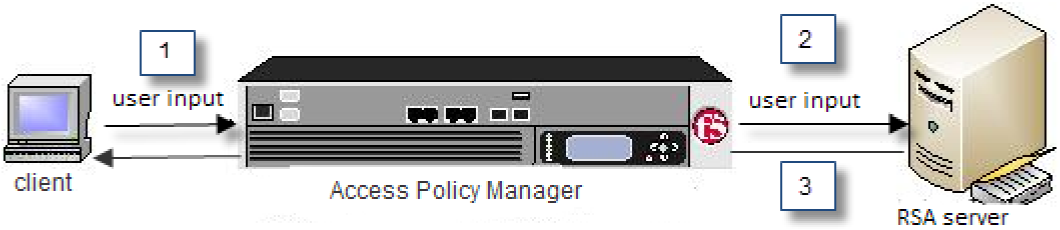


Figure 7: How Access Policy Manager works with RSA SecurID

1. Access Policy Manager® displays the logon page to the user. The logon page accepts the username and PIN code provided by the user, requests the passcode from RSA Secure-ID Software Token software, and sends the username and passcode to Access Policy Manager.
2. Access Policy Manager sends the user-specified inputs to the RSA authentication server.
3. Based on the authentication results, Access Policy Manager grants or denies access to the client.

About RSA SecurID (with soft token) automation requirements

For BIG-IP® Edge Client ® for Windows or BIG-IP Edge Client for Mac to support RSA SecurID (with soft token) automation, RSA SecurID must be configured for pin plus tokencode.

Task summary for configuring for RSA SecurID integration with APM

This task list includes all steps required to set up this configuration, and provides an example access policy that uses RSA SecurID authentication for F5® BIG-IP® Edge Client®. It is only an example. If you are adding RSA SecurID authentication to an existing access policy, you do not need to create another access profile.

Task list

Configuring a SecurID AAA server in APM

Creating an access profile

Verifying log settings for the access profile

Configuring RSA SecurID authentication in an access policy

Creating a virtual server

Configuring a SecurID AAA server in APM

Before you configure a SecurID AAA server, you must create a configuration file on the RSA SecurID console side to connect a BIG-IP® system to an RSA server.

Configure a SecurID AAA server for Access Policy Manager® (APM®) to request RSA SecurID authentication from an RSA Manager authentication server.

1. On the Main tab, click **Access** > **Authentication**.
   The Authentication screen opens.
2. On the menu bar, click **AAA Servers By Type**, and select **SecurID**.
   The SecurID screen opens and displays the servers list.
3. Click **Create**.
   The New Server properties screen opens.
4. In the **Name** field, type a unique name for the authentication server.
5. For the **SecurID Configuration File** setting, browse to upload the configuration file from the RSA SecurID console.

   Consult your RSA Authentication Manager administrator to generate this file for you.
6. Click **Finished**.
   The new server displays on the list.

This adds a new RSA SecurID server to the AAA Servers list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. Type a name for the access profile.
4. From the **Profile Type** list, select one:

   - **SSL-VPN** - Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL** - Select to support LTM-APM and SSL-VPN access types.

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring RSA SecurID authentication in an access policy

Before you add RSA SecurID authentication to an access policy, you must have at least one AAA SecurID server configured in Access Policy Manager® (APM®). You must also create a configuration file on the RSA SecurID console side to connect a BIG-IP® system to an RSA server. You might need an AAA server configured for another type of authentication, depending on the number of authentication actions that you plan to add to this access policy.

You add RSA SecurID authentication to an access policy so that APM can request RSA SecurID authentication using the AAA SecurID server that you specify.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **RSA SecurID** and click **Add Item**.
   A properties popup screen opens.
8. From the **AAA Server** list in the properties popup screen, select the SecurID AAA server that you want to associate to the agent.
9. Set **Max Logon Attempts** to a value from 1 to 5.
10. Click **Save**.
    The properties screen closes and the policy displays.
11. Add a Variable Assign action before the Logon Page action.
    After you add the Variable Assign action, a Properties popup screen opens.
12. On the Properties screen, add an entry for a session variable:
    a) Click **Add new entry**.
       An **empty** entry appears in the Assignment table.
    b) Click the **change** link in the new entry.
       A popup screen opens.
    c) From the left-side list, select **Custom Variable** (the default), and type `session.logon.page.softToken.fieldId`.

       This contains the field name on the logon page, which accepts a PIN from the client user.
    d) From the right-side list, select **Text**, and type `password`.

`password` is the name of the field that is used for RSA Software Token authentication.

e) Click **Finished**.

The popup screen closes.

f) Click **Save**.
The properties screen closes and the policy displays.

13. Click the **(+)** icon anywhere in the access policy to add a new item.

***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.

14. On the Assignment tab, select **Resource Assign** and click **Add Item**.
A properties popup screen opens.
15. From the Network Access list in the properties popup screen, select the network connection for your remote connection.
16. Click **Save**.
The properties screen closes and the policy displays.
17. Add any other actions you require.
18. Click **Apply Access Policy** to save your configuration.

This adds RSA SecurID AAA authentication to the access policy.

Creating a virtual server

When creating a virtual server for an access policy, specify an IP address for a single host as the destination address.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
The Virtual Server List screen opens.
2. Click the **Create** button.
The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **HTTP Profile** list, select **http**.
7. If you use server SSL for this connection, from the **SSL Profile (Server)** list, select a server SSL profile.
8. If you use client SSL for this profile, from the **SSL Profile (Client)** list, select a client SSL profile.
9. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
10. From the **Connectivity Profile** list, select a connectivity profile.

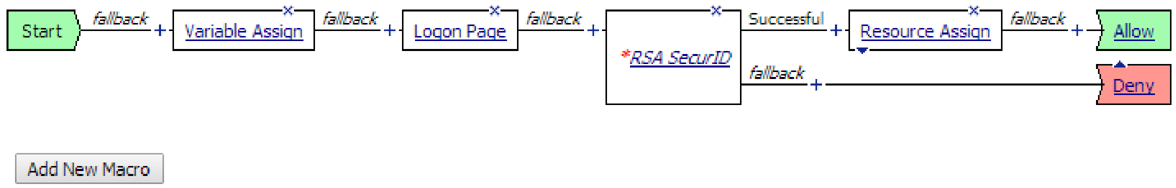
    You can select the default connectivity profile, **connectivity** if you have not defined a specific profile for the traffic that is directed to this virtual server.
11. Click **Finished**.

You have configured a host virtual server and associated an access profile with it.

Access policy example for RSA SecurID software token integration

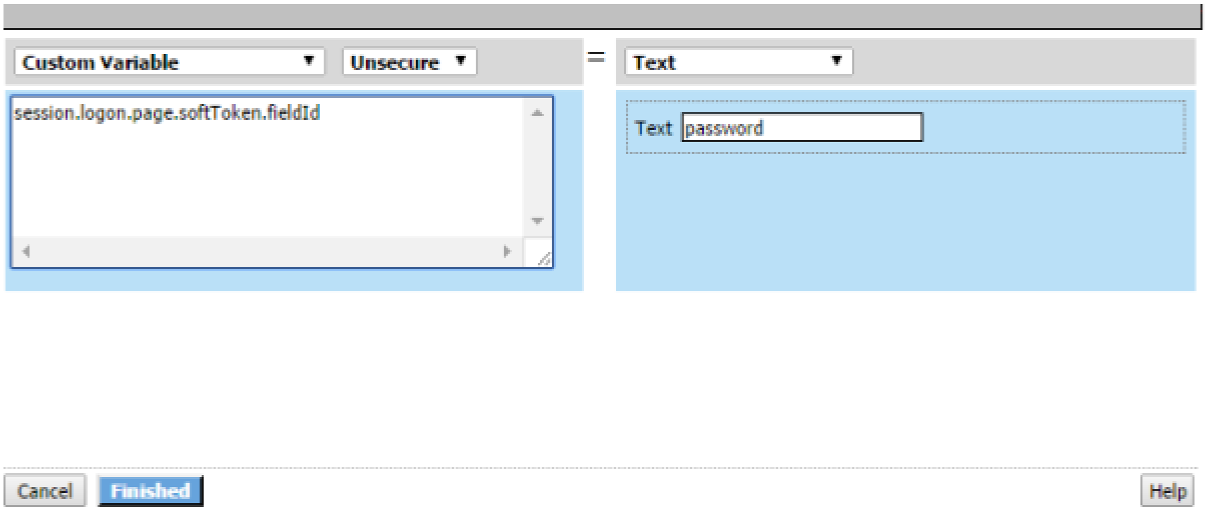
Typically, when you configure an authentication action, you precede it with a Logon Page action to collect credentials. This example describes how to include RSA SecurID integration in an access policy.

Access policy with RSA SecurID action



Variable Assign action

Use the Variable Assign action to provide the appropriate password before the Logon Page action occurs.


Custom Variable ▾ Unsecure ▾ = Text ▾

session.logon.page.softToken.fieldId

Text: password

The Variable Assign action stores the text `password` in the custom variable, `session.logon.page.softToken.fieldId`; `password` is the name of the field that is used for RSA token authentication.

RADIUS Authentication

About RADIUS authentication

Access Policy Manager® supports authenticating and authorizing the client against external RADIUS servers. When a client connects with the user name and password, Access Policy Manager authenticates against the external server on behalf of the client, and authorizes the client to access resources if the credentials are valid.

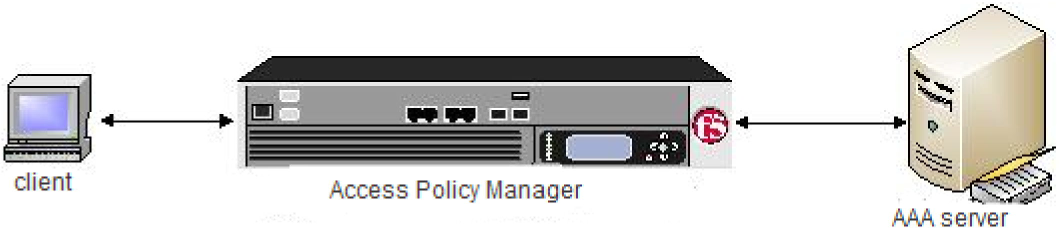


Figure 8: How RADIUS works

- The client requests access to network resources through Access Policy Manager.
- Access Policy Manager then issues a `RADIUS Access Request` message to the RADIUS server, requesting authorization to grant access.
- The RADIUS server then processes the request, and issues one of three responses to Access Policy Manager: `Access Accept`, `Access Challenge`, or `Access Reject`.

About AAA high availability

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established, as usual.

***Note:** Although new authentications fail if the BIG-IP® system loses connectivity to the server, existing sessions are unaffected provided that they do not attempt to re-authenticate.*

APM supports the following AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+. APM supports high availability by providing the option to create a pool of server connections when you configure the supported type of AAA server.

***Note:** If you use AAA with pools, such as RADIUS pools or Active Directory pools, APM assigns each pool member with a different number for the pool member's priority group value. APM must define each pool member with a different priority group because AAA load balancing is not used. The priority group number increases automatically with each created pool member. Alternative AAA pool configurations can be defined manually using the full flexibility of Local Traffic Manager™ (LTM®) if load balancing is desired.*

Guidelines for setting up RADIUS authentication for AAA high availability

When you use RADIUS as the authentication method for AAA high availability, there are general guidelines that you must follow when you set up your server connections.

- In a non-high availability environment, both the **Direct** and **Use Pool** options use the self IP address as a source IP address of the packet reaching the RADIUS server. For this scenario, you just need to add one IP address to the RADIUS allowed IP list to achieve this.
- In a high availability environment where the **Use Pool** option is used, the floating self IP address is used as a source IP of the RADIUS packet reaching the back-end. For this scenario, you need to add one self IP address (which is floating self IP address) to the RADIUS allowed IP list because the IP address is used even after a failover occurs.
- In a high availability environment where the **Direct** option is used, the self IP address is used as a source IP address of the RADIUS packet reaching the back-end. In this scenario, you need to add the self IP address from both active and standby devices to the RADIUS allowed IP list so that when failover occurs, the self IP address from the second device is accepted by the RADIUS server.

About how APM handles binary values in RADIUS attributes

For RADIUS authentication, Access Policy Manager® (APM®) converts an attribute value to hex if it contains unprintable characters, or if it is the `class` attribute. APM converts the class attribute to hex even if it contains only printable values (by attribute type). No other attributes are encoded to hex if they do not contain unprintable characters.

An attribute with a single unprintable value

```
1bf80e04.session.radius.last.attr.class 62 /
0x54230616000001370001ac1d423301caa87483dadf740000000000000007
```

Attribute with multiple values, both printable and unprintable (binary)

```
243be90d.session.radius.last.attr.class 119 0x6162636465666768696 /
a6b6c6d6e6f707172737475767778797a | 0x54220615000001370001ac1d423301caa87483 /
dadf740000000000000006
```

An attribute type that does not require hex encoding with both printable and unprintable values

```
3888eb70.session.radius.last.attr.login-lat-group 37 /
    0x6d7920bda12067726f757032 | mygroup1
```

In this case, only values that are unprintable are encoded to hex.

Task summary for RADIUS authentication

This task list includes all steps required to set up this configuration. If you add RADIUS authentication to an existing access policy, you already have an access profile configured and the access policy might already include a logon access policy item.

Task list

Configuring a RADIUS AAA server in APM

Creating an access profile

Verifying log settings for the access profile

Using RADIUS authentication in an access policy

Creating a virtual server

Configuring a RADIUS AAA server in APM

The Access Policy Manager ® (APM®) is a network access server (NAS) that operates as a client of the server configured here.

1. On the Main tab, click **Access** > **Authentication** > **RADIUS**.
   The RADIUS servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For the **Mode** setting, select **Authentication**.
5. For the **Server Connection** setting, select one of these options:

   Note: *When configuring a RADIUS AAA server that is located in a nondefault route domain, you must select* ***Use Pool*** *and specify the pool containing the RADIUS server.*

   - Select **Use Pool** to set up high availability for the AAA server.
   - Select **Direct** to set up the AAA server for standalone functionality.
6. If you selected **Use Pool**, type a name in the **Server Pool Name** field.

   You create a pool of servers on this screen.
7. Provide the addresses required for your server connection:
   - If you selected **Direct**, type an IP address in the **Server Address** field.
   - If you selected **Use Pool**, for each pool member you want to add, type an IP address in the **Server Addresses** field and click **Add**.

     Note: *When you configure a pool, you have the option to type the server address in route domain format:* `IPAddress%RouteDomain`.
8. In the **Authentication Service Port** field, type the authentication port number of your server. The default is `1812`.
9. In the **Secret** field, type the shared secret password of the server.
10. In the **Confirm Secret** field, re-type the shared secret password of the server.
11. Click **Finished**.
    The new server displays on the list.

The new AAA server displays on the RADIUS Servers list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Using RADIUS authentication in an access policy

You configure an access policy with a RADIUS Auth action to provide RADIUS authentication as one of authentication options for users trying to gain accesss.

Note: *You can use RADIUS authentication in addition to other authentication types. You can require that users pass at least one type of authentication or that they pass multiple types of authentication.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. From the Authentication tab, select **RADIUS Auth** and click **Add Item**.
   The popup screen closes. A Properties popup screen opens.

8. On the Properties popup screen from the **AAA Server** list, select the AAA RADIUS server you configured previously and click **Save**.
   The popup screen closes and the visual policy editor displays.
9. Complete the policy:
   a) Add any additional policy items you require.
   b) Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.
10. Click **Apply Access Policy** to save your configuration.

This creates an access policy that collects user credentials and uses them to authenticate with a RADIUS server.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Creating a virtual server

When creating a virtual server for an access policy, specify an IP address for a single host as the destination address.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **HTTP Profile** list, select **http**.
7. If you use server SSL for this connection, from the **SSL Profile (Server)** list, select a server SSL profile.
8. If you use client SSL for this profile, from the **SSL Profile (Client)** list, select a client SSL profile.
9. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
10. From the **Connectivity Profile** list, select a connectivity profile.

    You can select the default connectivity profile, **connectivity** if you have not defined a specific profile for the traffic that is directed to this virtual server.
11. Click **Finished**.

You have configured a host virtual server and associated an access profile with it.

Testing AAA high availability for supported authentication servers

To effectively test that high availability works for your authentication servers, you should have two servers that are accessible, where you can remove one of them from the network.

Note: *High availability is supported for these authentication server types only: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.*

If you configured a supported authentication server type to use a pool of connection servers, you can test the configuration using these steps.

1. Begin a `tcpdump` on the Access Policy Manager®, using a protocol analyzer, and scanning for packets destined for the specific port for your authentication server.
2. Log in to the virtual server with both servers active.
3. Using the `tcpdump` records, verify that the requests are being sent to the higher priority server.
4. Log out of the virtual server.
5. Disable the higher-priority server.
6. Log in to the virtual server again.
7. Verify that the request is being sent to the other server.
8. Log out again, re-enabling the server, and try one more time to verify that the new requests are being sent to the high priority server.

RADIUS attributes

The following table lists the specific RADIUS attributes that Access Policy Manager® sends with RADIUS requests.

Attribute	Purpose
`User-Name`	Indicates the name of the authenticated user.
`User-Password`	Indicates the password of the authenticated user.
`NAS-IP-Address`	Indicates the identifying IP Address of the NAS.
`NAS-IPv6-Address`	Indicates the identifying IPv6 Address of the NAS.
`NAS-Identifier`	Indicates the identifying name of the NAS .
`Service-Type`	Indicates the type of service the user has requested.
`NAS-Port`	Indicates the physical port number of the NAS that is authenticating the user.

RADIUS session variables for access policy rules

When the RADIUS Auth access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the RADIUS authentication access policy item and for a logon access policy item.

Session variables for RADIUS

Session Variable	Description
`session.RADIUS.last.result`	Provides the result of the RADIUS authentication. The available values are: • 0: Failed • 1: Passed
`session.RADIUS.last.attr.$attr_name`	*`$attr_name`* is a value that represents the user's attributes received during RADIUS authentication. Each attribute is converted to separate session variables.

Session Variable	Description
`session.RADIUS.last.errmsg`	Displays the error message for the last login. If `session.RADIUS.last.result` is set to 0, then `session.RADIUS.last.errmsg` might be useful for troubleshooting purposes. Example: `c76a50c0.session.RADIUS.last.errmsg 13 Access-Reject`

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

RADIUS authentication and accounting troubleshooting tips

You might run into problems with RADIUS authentication and accounting in some instances. Follow these tips to try to resolve any issues you might encounter.

RADIUS authentication and accounting access policy action troubleshooting

Possible error messages	Possible explanations and actions
`Authentication failed due to timeout`	• Verify that Access Policy Manager® is configured as a client on the RADIUS server. • You might have encountered a general network connection problem.
`Authentication failed due to RADIUS access reject`	• Verify that the shared secret on the RADIUS server is valid. • Verify that user credentials are entered correctly.

Additional troubleshooting tips for RADIUS authentication and accounting

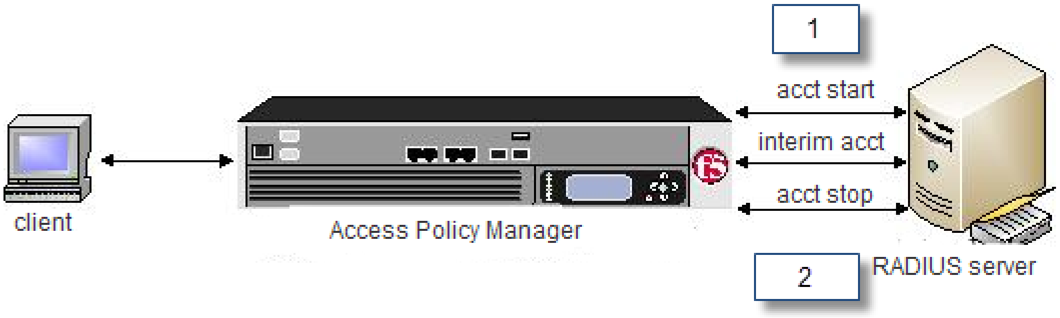
Action	Steps
Check to see if your access policy is attempting to perform authentication	• Add message boxes to your access policy to display information about what the access policy is attempting to do. • Refer to `/var/log/apm` to view authentication and accounting attempts by the access policy. ***Note:*** *Make sure that your log level is set to the appropriate level. The default log level is* `notice`.
Check the RADIUS Server configuration	• Confirm that the Access Policy Manager is registered as a RADIUS client. Since the Access Policy Manager makes requests from the self IP address to the RADIUS server for authentication requests, the

Action	Steps
	address of the self-IP address should be registered as a RADIUS client. • Check the RADIUS logs and check for any errors.
Confirm network connectivity	• Access the BIG-IP® system through the command line interface and check your connectivity by pinging the RADIUS server using the host entry in the AAA Server box. • Confirm that the RADIUS port 1812 is not blocked between the Access Policy Manager and the RADIUS server.
Capture a TCP dump	• Take a TCP dump from the Access Policy Manager when authentication attempts are made. For example, %TCP dump-i 1.1 -s /tmp/dump. You must first determine what interface the self IP address is on. These TCP dumps indicate activities between the Access Policy Manager and the authentication server. • Run the authentication test. After authentication fails, stop the TCP dump, download the TCP dump records to a client system, and use an analyzer to troubleshoot. ***Important:*** *If you decide to escalate the issue to customer support, you must provide a capture of the TCP dump when you encounter authentication issues that you cannot otherwise resolve on your own.*

RADIUS Accounting

About RADIUS accounting

You can report user session information to an external RADIUS accounting server. If you select this mode only, the system assumes that you have set up another type of authentication method to authenticate and authorize your users to access their resources.



1. After RADIUS accounting runs successfully in an access policy, Access Policy Manager® sends an accounting start request message to the external RADIUS server. The `start` message typically contains the user's ID, networks address, point of attachment, and a unique session identifier.
2. When the session is destroyed, Access Policy Manager issues an accounting `stop` message to the external RADIUS server, providing information on the final usage in terms of time, packets transferred, data transferred, and reason for disconnect, as well as other information related to the user's access.

This accounting data is used primarily for billing, statistical, and general network monitoring purposes.

***Note:** You can perform both RADIUS authentication and accounting actions. Keep in mind that if you select this mode, the RADIUS server and the RADIUS accounting server must run on different service ports.*

About how APM handles binary values in RADIUS attributes

For RADIUS authentication, Access Policy Manager® (APM®) converts an attribute value to hex if it contains unprintable characters, or if it is the `class` attribute. APM converts the class attribute to hex even if it contains only printable values (by attribute type). No other attributes are encoded to hex if they do not contain unprintable characters.

An attribute with a single unprintable value

```
1bf80e04.session.radius.last.attr.class 62 /
0x54230616000001370001ac1d423301caa87483dadf740000000000000007
```

Attribute with multiple values, both printable and unprintable (binary)

```
243be90d.session.radius.last.attr.class 119 0x6162636465666768696 /
a6b6c6d6e6f707172737475767778797a | 0x54220615000001370001ac1d423301caa87483 /
dadf74000000000000000006
```

An attribute type that does not require hex encoding with both printable and unprintable values

```
3888eb70.session.radius.last.attr.login-lat-group 37 /
    0x6d7920bda12067726f757032 | mygroup1
```

In this case, only values that are unprintable are encoded to hex.

Configuring a RADIUS Accounting server in APM

1. On the Main tab, click **Access** > **Authentication** > **RADIUS**.
   The RADIUS servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. From the **Mode** list, select **Accounting**.
5. For the **Server Connection** setting, select one of these options:
   - Select **Use Pool** to set up high availability for the AAA server.
   - Select **Direct** to set up the AAA server for standalone functionality.
6. If you selected **Use Pool**, type a name in the **Server Pool Name** field.

   You create a pool of servers on this screen.
7. Provide the addresses required for your server connection:
   - If you selected **Direct**, type an IP address in the **Server Address** field.
   - If you selected **Use Pool**, for each pool member you want to add, type an IP address in the **Server Addresses** field and click **Add**.

   Note: *When you configure a pool, you have the option to type the server address in route domain format:* `IPAddress%RouteDomain`.
8. If you selected **Use Pool**, you have the option to select a **Server Pool Monitor** to track the health of the server pool.
9. In the **Accounting Service Port** field, type the service port for your accounting server if the default value is not appropriate.

   The default is `1813`.
10. In the **Secret** field, type the shared secret password of the server.
11. In the **Confirm Secret** field, re-type the shared secret password of the server.
12. In the **Timeout** field, type a timeout interval (in seconds) for the AAA server.

    This setting is optional.

    If you use the **Timeout** setting, you can also use the **Retries** setting. If these settings are enabled, the Access Policy Manager attempts to reach the AAA server within the specified time frame, in seconds.

If the server does not respond, the Access Policy Manager retries the authentication attempt, depending on how many retries you specify.

13. In the **Retries** field, type the number of times the BIG-IP system should try to make a connection to the server after the first attempt fails.

    This setting is optional.
14. Click **Finished**.
    The new server displays on the list.

Adding RADIUS accounting to an access policy

Before you set up an access policy to use RADIUS accounting, you must have at least one RADIUS AAA server configured. You should also have an access profile that is configured with actions that authenticate the user.

You add a RADIUS accounting action to an access policy to send RADIUS start and stop messages to a RADIUS server. RADIUS accounting does not authenticate a user.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. From the Authentication tab, select **RADIUS Acct** and click **Add Item**.
   The popup screen closes. A properties popup screen opens.
5. From the **AAA Server** list, select a RADIUS accounting server and click **Save**.
   The properties popup screen closes and the visual policy editor displays.
6. Click **Apply Access Policy** to save your configuration.

This adds the RADIUS accounting action to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.

2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

RADIUS authentication and accounting troubleshooting tips

You might run into problems with RADIUS authentication and accounting in some instances. Follow these tips to try to resolve any issues you might encounter.

RADIUS authentication and accounting access policy action troubleshooting

Possible error messages	Possible explanations and actions
`Authentication failed due to timeout`	• Verify that Access Policy Manager® is configured as a client on the RADIUS server. • You might have encountered a general network connection problem.
`Authentication failed due to RADIUS access reject`	• Verify that the shared secret on the RADIUS server is valid. • Verify that user credentials are entered correctly.

Additional troubleshooting tips for RADIUS authentication and accounting

Action	Steps
Check to see if your access policy is attempting to perform authentication	• Add message boxes to your access policy to display information about what the access policy is attempting to do. • Refer to `/var/log/apm` to view authentication and accounting attempts by the access policy. ***Note:*** *Make sure that your log level is set to the appropriate level. The default log level is* `notice`*.*
Check the RADIUS Server configuration	• Confirm that the Access Policy Manager is registered as a RADIUS client. Since the Access Policy Manager makes requests from the self IP address to the RADIUS server for authentication requests, the address of the self-IP address should be registered as a RADIUS client. • Check the RADIUS logs and check for any errors.

Action	Steps
Confirm network connectivity	• Access the BIG-IP® system through the command line interface and check your connectivity by pinging the RADIUS server using the host entry in the AAA Server box. • Confirm that the RADIUS port 1812 is not blocked between the Access Policy Manager and the RADIUS server.
Capture a TCP dump	• Take a TCP dump from the Access Policy Manager when authentication attempts are made. For example, %TCP dump-i 1.1 -s /tmp/dump. You must first determine what interface the self IP address is on. These TCP dumps indicate activities between the Access Policy Manager and the authentication server. • Run the authentication test. After authentication fails, stop the TCP dump, download the TCP dump records to a client system, and use an analyzer to troubleshoot. **Important:** *If you decide to escalate the issue to customer support, you must provide a capture of the TCP dump when you encounter authentication issues that you cannot otherwise resolve on your own.*

Kerberos Authentication with End-User Logons

About basic authentication and Kerberos end-user logon

Access Policy Manager® (APM®) provides an alternative to a form-based login authentication method. This alternative method uses a browser login box that is triggered by an HTTP 401 response to collect credentials. A SPNEGO/Kerberos or basic authentication challenge can generate a HTTP 401 response.

This option is useful when a user is already logged in to the local domain and you want to avoid submitting an APM HTTP form for collecting user credentials. The browser automatically submits credentials to the server and bypasses the login box to collect the credentials again.

***Note:** Because SPNEGO/Kerberos is a request-based authentication feature, the authentication process is different from other authentication methods, which run at session creation time. SPNEGO/Kerberos authentication can occur at any time during the session.*

The benefits of this feature include:

- Provides flexible login mechanism instead of restricting you to use only the form-based login method.
- Eliminates the need for domain users to explicitly type login information again to log in to Access Policy Manager.
- Eliminates the need for user password transmission with Kerberos method.

***Important:** Administrators should not turn off the **KeepAlive** setting on the web server because turning that setting off might interfere with Kerberos authentication.*

How does end-user logon work?

To retrieve user credentials for end-user logon, you can use basic authentication or SPEGNO/Kerberos methods or both.

Basic authentication

Use this method to retrieve user credentials (user name and password) from a browser. You can think of this method as a replacement for form-based authentication used by the standard login screen. If you use basic authentication, the system populates the user name and password session variables, which can then be used by any other authentication actions, such as Active Directory or RADIUS.

SPNEGO/Kerberos

Use this method to retrieve user credentials through SPNEGO/Kerberos authentication header. With the Kerberos method, the client system must first join a domain and a Kerberos action must follow. The Kerberos action does not run immediately; it runs only when clients request SPNEGO/Kerberos authentication. By default, Kerberos authentication runs not only on the first request, but also on subsequent requests where authentication is needed, such as for new connections. Access Policy Manager® (APM®) validates the request by confirming that a valid ticket is present.

***Note:** You can disable Kerberos per request-based authentication in the Kerberos authentication access policy item configuration in APM. If you disable it, authentication occurs while the access policy runs and subsequent authentications do not occur. In that case, end-user logon does not occur.*

***Note:** You can achieve multi-domain support for Kerberos authentication through multiple virtual servers. Each virtual server must have its own access policy and its own Kerberos configuration.*

Both methods require that an HTTP 401 Response action item be configured in the access policy and that the authentication method be specified in the action item. In cases where both methods are selected, the browser determines which method to perform based on whether the system has joined a domain. The HTTP 401 Response action has two default branches to indicate whether basic authentication or Kerberos method is performed.

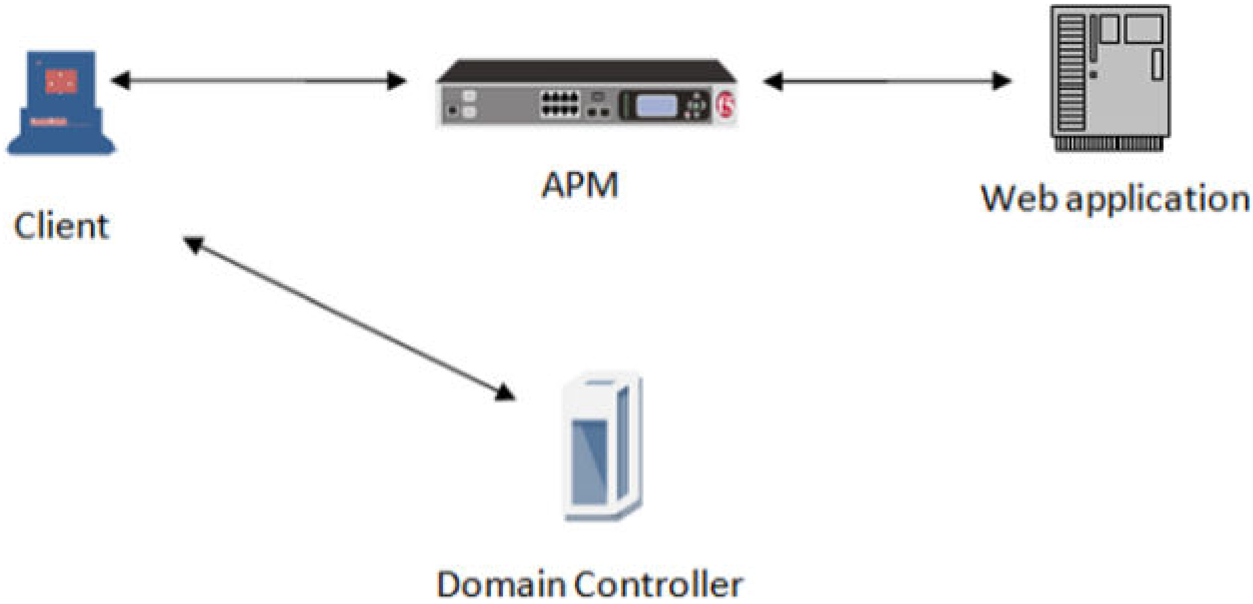


Figure 9: How SPNEGO/Kerberos end-user logon works

The end-user logon works with events happening in this order:

- The client becomes a member and connects to the domain.
- The client connects to a virtual server on the BIG-IP® system.
- The access policy runs and issues a 401 HTTP request action.
- If Kerberos is present, the browser forwards the Kerberos ticket along with the request when it receives the 401 HTTP request.
- Access Policy Manager validates the Kerberos ticket after the request is received and determines whether or not to permit the request.

About Kerberos authentication requirements

To configure Kerberos authentication, you must meet specific configuration requirements as described here.

Virtual server

The virtual server IP address and host name are necessary to configure DNS.

DNS configuration

Make sure you have the zone file and PTR record for the virtual server IP address. For example:

```
testbed.lab.companynet 10.10.4.100
```

Browser configuration

Configure the browser to use Kerberos. Typically, Internet Explorer is already configured for Kerberos; however, you might need to configure it for trusted sites. To use Firefox, you must configure it for negotiate authentication.

Task summary for configuring end-user login support

To set up this configuration, perform the procedures in the task list.

Task list

Joining a Kerberos user account to a domain
Configuring an AAA server for Kerberos authentication
Creating an access profile
Verifying log settings for the access profile
Configuring an access policy for end-user logon support

Joining a Kerberos user account to a domain

To use Kerberos authentication, you need the client joined and connected to a domain and you need a keytab file.

1. Create a surrogate user in the domain.

   In this example, the hostname of the virtual server on the BIG-IP® system is testbed.lab.companynet.com and the user name is john.

   ```
   setspn -U -A HTTP/testbed.lab.companynet.com john
   ```
2. Map the user account to the service account and generate a keytab file for the service.

   You can use the ktpass utility to do this.

   In this example, `LAB.COMPANYNET.COM` specifies the Kerberos authentication realm, which is case-sensitive and must be specified in uppercase. The user name, which is specified in UPN format (`john@lab.companynet.com`), is not case-sensitive. (The user name can be specified in user name format, DN format, or UPN format).

   ```
   c:>ktpass -princ HTTP/testbed.lab.companynet.com@LAB.COMPANYNET.COM -
   mapuser john@lab.companynet.com -crypto rc4-hmac-nt -ptype KRB5_NT_SRV_HST
   -pass password -out c:\temp\john.keytab
   ```

Configuring an AAA server for Kerberos authentication

Configure a Kerberos AAA server so that you can add it to a Kerberos authentication action in an access policy.

1. On the Main tab, click **Access** > **Authentication** > **Kerberos**.
   The Kerberos Servers list screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. In the **Auth Realm** field, type a Kerberos authentication realm name (administrative name), such as `LAB.COMANYNET`.

   Type the realm name all uppercase; it is case-sensitive.
5. In the **Service Name** field, type a service name; for example, `HTTP`.
6. In the **Keytab File** area, click **Choose File** to locate and upload the keytab file.

   A keytab file contains Kerberos encryption keys (these are derived from the Kerberos password).
7. Click **Finished**.
   The new server displays on the list.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy for end-user logon support

To use basic authentication in addition to Kerberos authentication, you need an AAA server configured for the authentication agent that you plan to use.

Configure an access policy like this one to handle basic and SPEGNO/Kerberos authentication challenges without submitting an Access Policy Manager® HTTP form to collect user credentials.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. Under General Purpose, select **HTTP 401 Response**, and click **Add item**.
   A properties screen opens.
5. In the 401 Response Setting area from the **HTTP Auth Level** list, select **basic+negotiate**, and click **Save**.
   The properties screen closes. The visual policy editor displays the HTTP 401 Response item with 3 branches: Basic, Negotiate, and fallback.
6. To perform basic authentication, add an authentication server agent on the **Basic** branch.
7. To use the Kerberos authentication method:
   a) Add the **Kerberos Auth** agent on the **Negotiate** branch.
      After you add the Kerberos Auth item, a properties popup screen displays.
   b) On the properties screen for the **AAA Server** setting, select the Kerberos AAA server.
   c) Click **Save**.
      The properties screen closes and the policy displays.

8. Complete the policy:
   a) Add any additional policy items you require.
   b) Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.
9. Click **Apply Access Policy**.

To apply this access policy to network traffic, add the access profile to a virtual server.

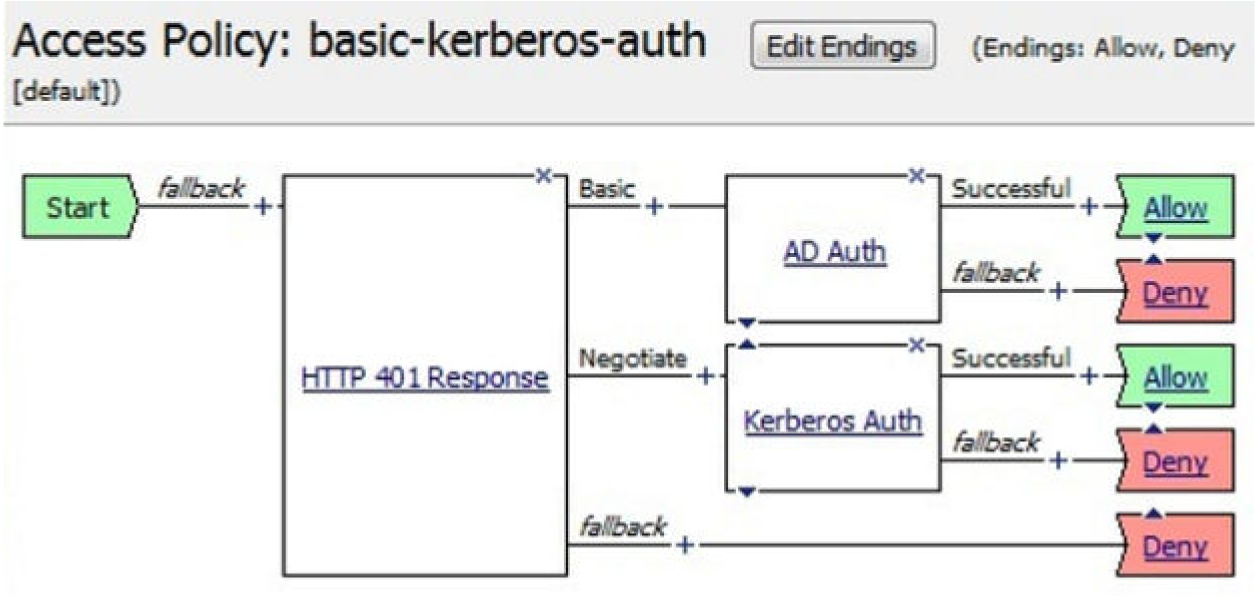
Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Access policy example for end-user login

This is an example of an access policy with all the associated elements needed to successfully support the end-user login feature. Notice that separate branches are created automatically to support using either basic authentication or Kerberos method to retrieve user credentials.

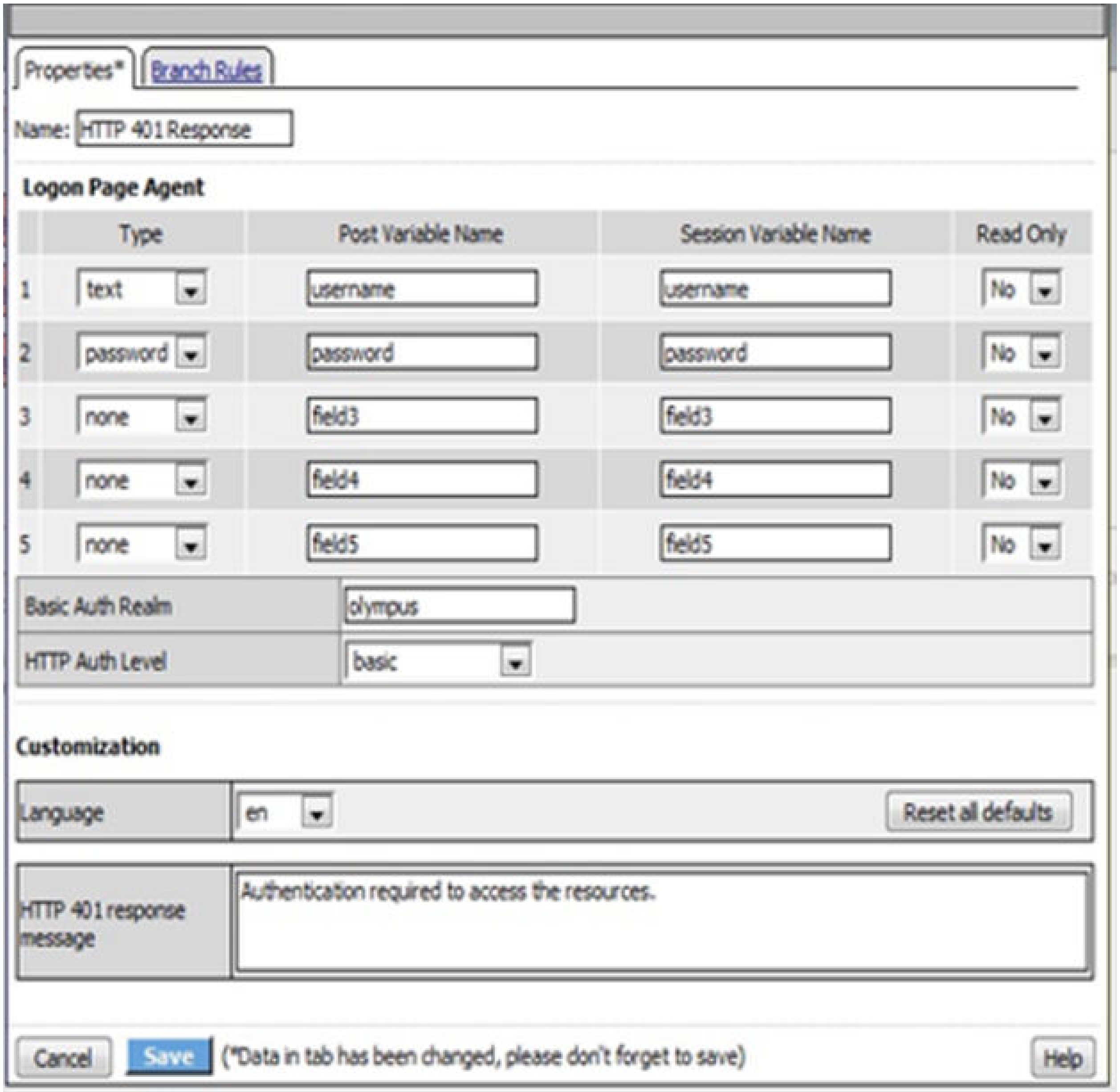
Note: *For basic authentication, the user name and password validation occurs at the session creation time. After the access policy completes, the session cookie is used to validate the session.*

Note: *By default, Kerberos runs not only at the access policy run time but also at any time in the session.*



Access Policy: basic-kerberos-auth

Figure 10: Example access policy for end-user login



Name: HTTP 401 Response

Logon Page Agent

	Type	Post Variable Name	Session Variable Name	Read Only
1	text	username	username	No
2	password	password	password	No
3	none	field3	field3	No
4	none	field4	field4	No
5	none	field5	field5	No

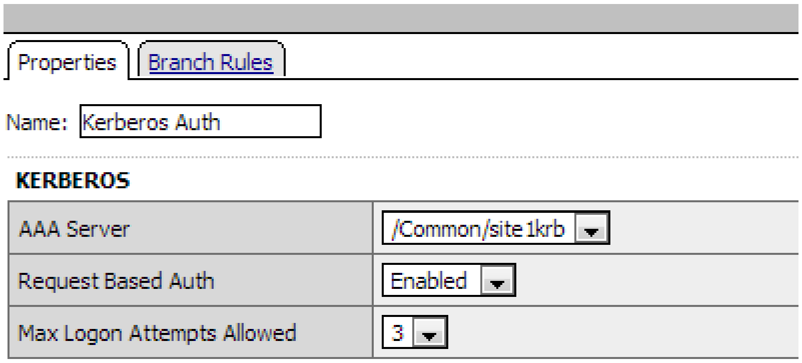
Basic Auth Realm	olympus
HTTP Auth Level	basic

Customization

Language	en
HTTP 401 response message	Authentication required to access the resources.

(*Data in tab has been changed, please don't forget to save)

Figure 11: Example properties for an HTTP 401 response action



Name: Kerberos Auth

KERBEROS

AAA Server	/Common/site1krb
Request Based Auth	Enabled
Max Logon Attempts Allowed	3

Figure 12: Example properties for a Kerberos Auth action on the Negotiate branch

Kerberos authentication troubleshooting tips

You might choose to verify Kerberos authentication configurations in some instances. Use these troubleshooting tips to help resolve any issues you might encounter.

Verify the keytab file

From the command line, use the `klist` command as shown in this example.

Important: *The command must be typed on one line.*

```
klist -ke WRFILE:/config/filestore/files_d/Common_d/kerberos_keytab_file_d/\:Common\:SUN-
SPNEGO-APM106_key_file_2
```

The output for the example contains information like this.

```
Keytab name:
FILE:/config/filestore/files_d/Common_d/kerberos_keytab_file_d/:Common:SUN-SPNEGO-
APM106_key_file_2
KVNO Principal
3    HTTP/apm106.labt.companynet.com@labt.companynet.com(arcfour-hmac)
```

Verify Kerberos delegation

From the command line, use the `kinit` command, as shown in this example.

```
kinit HTTP/apm106.labt.companynet.com@labt.companynet.com
```

You are prompted for a password and should receive a ticket (no output, no error).

Verify ticket

From the command line, type `klist`. Here is sample output: `/etc/krb5.conf`

Capture a TCP dump

Make sure the client sends the ticket to the BIG-IP® system; this verifies that the client setup is successful.

NTLM Authentication for Microsoft Exchange Clients

Overview: Configuring APM for Exchange clients that use NTLM authentication

Access Policy Manager® (APM®) supports Microsoft Exchange clients that are configured to use NTLM, by checking NTLM outside of the APM session as needed. APM requires a machine account and an NTLM Auth configuration to perform these checks. APM requires an Exchange profile to support Microsoft Exchange clients, regardless of the authentication they are configured to use.

Task summary

Configuring a machine account
Creating an NTLM Auth configuration
Setting up a delegation account to support Kerberos SSO
Creating a Kerberos SSO configuration in APM
Configuring an Exchange profile
Creating an access profile for Exchange clients
Verifying log settings for the access profile
Configuring an access policy for NTLM authentication
Adding the access profile to the virtual server
Maintaining a machine account

About using NTLM authentication

Microsoft software systems use NTLM as an integrated single sign-on (SSO) mechanism. However, in an Active Directory-based SSO scheme, Kerberos replaces NTLM as the default authentication protocol. NTLM is still used when a domain controller is not available or is unreachable, such as when the client is not Kerberos-capable, the server is not joined to a domain, or the user authenticates remotely over the web.

About configuration requirements for NTLM authentication

In Access Policy Manager®, you need to configure these elements:

- Machine account
- NTLM authentication configuration
- Kerberos SSO configuration
- Exchange profile that specifies the NTLM authentication configuration and specifies Kerberos SSO configurations for the specific Microsoft Exchange services supported
- Access profile that specifies the Exchange profile
- Access policy
- Pool of servers for the Exchange service to support Outlook Anywhere, supply a pool of Outlook Anywhere servers
- Virtual server that specifies the access profile and the pool

You also need to configure a special account in Active Directory for Kerberos constrained delegation (KDC).

About reusing a machine account for different BIG-IP systems

You can use the same machine account for two BIG-IP® systems when they are in an active-standby configuration. Otherwise, F5® recommends that you create a new NTLM machine account using the Access Policy Manager® user interface on each BIG-IP system.

Creating a new NTLM machine account on each BIG-IP system is helpful, for example, when two systems independently update their configurations without propagating them, or when you replicate the configuration into different BIG-IP systems using any configuration replication method. If you export a configuration and import it on another system, the machine account is included; however, after the import completes, you still need a new machine account and an NTLM authentication configuration that uses the new machine account on the target system.

About Outlook Anywhere and NTLM authentication

Access Policy Manager® (APM®)supports Outlook Anywhere clients that are configured to use NTLM and HTTP Basic protocols independently. Typically, mobile devices use HTTP Basic authentication, while Outlook Anywhere clients can use both NTLM and HTTP Basic authentication. APM determines whether a client uses NTLM or HTTP Basic authentication and enforces the use of one or the other. After a client authenticates with NTLM or HTTP Basic, APM supports single sign-on with the back-end application or server using Kerberos constrained delegation (KCD).

Configuring a machine account

You configure a machine account so that Access Policy Manager® (APM®) can establish a secure channel to a domain controller.

1. On the Main tab, click **Access** > **Authentication** > **NTLM** > **Machine Account**.
   A new Machine Account screen opens.
2. In the Configuration area, in the **Machine Account Name** field, type a name.
3. In the **Domain FQDN** field, type the fully qualified domain name (FQDN) for the domain that you want the machine account to join.
4. (Optional) In the **Domain Controller FQDN** field, type the FQDN for a domain controller.
5. In the **Admin User** field, type the name of a user who has administrator privilege.
6. In the **Admin Password** field, type the password for the admin user.

   APM uses these credentials to create the machine account on the domain controller. However, APM does not store the credentials and you do not need them to update an existing machine account configuration later.
7. Click **Join**.

This creates a machine account and joins it to the specified domain. This also creates a non-editable **NetBIOS Domain Name** field that is automatically populated.

Note: *If the* ***NetBIOS Domain Name*** *field on the machine account is empty, delete the configuration and recreate it. The field populates.*

Creating an NTLM Auth configuration

Create an NTLM Auth configuration to specify the domain controllers that a machine account can use to log in.

1. On the Main tab, click **Access** > **Authentication** > **NTLM** > **NTLM Auth Configuration**.
   A new NTLM Auth Configuration screen opens.

2. In the **Name** field, type a name.
3. From the **Machine Account Name** list, select the machine account configuration to which this NTLM Auth configuration applies.

   You can assign the same machine account to multiple NTLM authentication configurations.
4. For each domain controller, type a fully qualified domain name (FQDN) and click **Add**.

   Note: *You should add only domain controllers that belong to one domain.*

   By specifying more than one domain controller, you enable high availability. If the first domain controller on the list is not available, Access Policy Manager® tries the next domain controller on the list, successively.
5. Click **Finished**.

This specifies the domain controllers that a machine account can use to log in.

Setting up a delegation account to support Kerberos SSO

Before you can configure Kerberos SSO in Access Policy Manager®, you must create a delegation account in Active Directory.

Note: *For every server realm, you must create a delegation account in that realm.*

1. Open the Active Directory Users and Computers administrative tool and create a new user account.

   The user account should be dedicated for delegation, and the **Password never expires** setting enabled.
2. Set the service principal name (SPN) on the Windows server for the user account.

   For the support tools that you can use, and for the commands, such as `setspn` and `ktpass`, refer to Microsoft documentation.

   Note: *If you use the* `ktpass` *command, it sets the SPN on the Windows server and creates a keytab file. APM Kerberos SSO does not need or use a keytab file.*
3. Verify the result of setting the SPN.

   This example is purely for illustration. Refer to Microsoft documentation for up-to-date commands and correct usage.

   ```
   C:\Users\Administrator> setspn -L apm4
   ```

   ```
   Registered ServicePrincipalNames for
   CN=apm4,OU=users,DC=yosemite,DC=lab,DC=dnet,DC=com: HTTP/
   apm4.yosemite.lab.dnet.com
   ```
   where *`apm4`* is the name of the user account that you created.
4. Return to the Active Directory Users and Computers screen to open your account again.

   A Delegation tab should appear.
5. Click the Delegation tab.
6. Select **Trust this user for delegation to specified services only**.
7. Select **Use any authentication protocol**, and add all your services to the list under **Services to which this account can present delegated credentials**.

   Every service should have Service Type HTTP (or http) and host name of the pool member or web application resource host that you will use in your configuration.
8. Click **OK**.
   This creates the new delegation account.

Creating a Kerberos SSO configuration in APM

Before you start, you must have configured a delegation account in Active Directory.

To support Kerberos single sign-on authentication from Access Policy Manager® (APM®), you must create a Kerberos SSO configuration.

Note: *To complete this task, you need to know the service principal name (SPN) for the delegation account.*

1. On the Main tab, click **Access** > **Single Sign-On** > **Kerberos**.
   The Kerberos screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
6. In the **Kerberos Realm** field, type the name of the realm in uppercase.

   For example, `MY.HOST.LAB.MYNET.COM`
7. In the **Account Name** field, type the name of the Active Directory account configured for delegation.

   Type the account name in SPN format.

   In this example `HTTP/apm4.my.host.lab.mynet.com@MY.HOST.LAB.MYNET.COM`, apm4 is the delegation account, apm4.my.host.lab.mynet.com is its fully qualified domain name, and MY.HOST.LAB.MYNET.COM is the realm.
8. In the **Account Password** and **Confirm Account Password** fields, type the delegation account password.
9. Click **Finished**.

Configuring an Exchange profile

If any of the Microsoft Exchange clients you support authenticate using NTLM, you must first create these objects:

- A machine account
- An NTLM Auth configuration
- At least one Kerberos SSO configuration

Note: *For Access Policy Manager® (APM®) to support Kerberos SSO, a delegation account is required on Active Directory.*

You create an Exchange profile to specify how to handle traffic from Microsoft Exchange clients.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Microsoft Exchange**.
   A list of Exchange profiles displays.
2. Click **Create**.
   A Create New Exchange Profile popup screen displays general settings.
3. In the **Exchange Name** field, type a name for the Exchange profile.
4. From the **Parent Profile** list, select a profile.

The Exchange profile inherits settings from the parent profile that you select.

***Note:** APM supplies a default Exchange profile named exchange.*

5. Repeat these steps for one or more Microsoft Exchange services:
   a) From Service Settings on the left, select an Exchange service.
   Settings for the service are displayed in the right pane.
   b) In the **URL** field, retain any default settings that are displayed or type a path to use to match the Exchange client.

      Default settings for this field are supplied in the default exchange profile.
   c) From the **Front End Authentication** list, select the type of authentication to use: **Basic**, **Basic-NTLM**, or **NTLM**.

      Only the applicable authentication types for the particular the Exchange service are included on the list.

      ***Note:** If you select **NTLM** or **Basic-NTLM**, you must also select a configuration from **NTLM Configuration** list on the General Settings screen.*
   d) From the **SSO Configuration** list, select an SSO configuration, if needed, for use after initial login.

      For **Basic-NTLM**and **NTLM** authentication types, only Kerberos SSO is supported.

   You configured settings for one or more Microsoft Exchange services.
6. Click **OK**.
   The screen closes.

The Exchange profile is displayed on the list.

Apply this Exchange profile by adding it to an access profile.

Creating an access profile for Exchange clients

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session. You add an Exchange profile to the access profile to specify how to handle traffic from Microsoft Exchange clients.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   ***Note:** A access profile name must be unique among all access profile and any per-request policy names.*
4. (Optional) In the Configurations area from the **Exchange** list, select an Exchange profile.

   Exchange profiles specify any SSO configurations for Microsoft Exchange services, such as Autodiscover, Outlook Anywhere, and so on. The configuration in the Exchange profile is used for Microsoft Exchange clients regardless of any SSO configuration you select from the **SSO Configuration** list in this access profile.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

7. To change from using the default-log-settings that APM automatically adds to the access profile, you can do this.:

   Logging occurs for a session only when a log setting is specified for the access profile.

   a) Click the name of the access profile.
      The Properties screen opens.
   b) On the menu bar, click **Logs**.
      The General Properties screen opens.
   c) In the Log Settings area, move log settings from the **Available** list to the **Selected** list.
   d) Click **Update**.

   You can configure log settings in the **Access** > **Overview** > **Event Log** > **Settings** area of the product.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy for NTLM authentication

You configure an access policy for NTLM authentication to support Outlook Anywhere clients that log in using NTLM to also gain SSO access to a backend server that is protected by Kerberos KCD.

Note: *NTLM authentication occurs before an access policy runs. If NTLM authentication fails, an error displays and the access policy does not run.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.

4. On the Endpoint Security (Server-Side) tab, select **Client for MS Exchange** and click **Add Item** to add the action to the access policy.

   A Client for MS Exchange action determines whether the client is using Microsoft Exchange or ActiveSync protocols. You must add this action before an NTLM Auth Result action.

   The Client for MS Exchange action popup screen opens.
5. Click **Save**.
   The properties screen closes and the policy displays.
6. Check whether the Outlook Anywhere client authenticated using NTLM.
   a) Click the **[+]** sign on the successful branch after the Client for MS Exchange action.
   An Add Item window opens.
   b) On the **Authentication** tab, select **NTLM Auth Result**.
   c) Click **Add Item**.
   A popup screen opens.
   d) Click **Save**.
   The properties screen closes and the policy displays.
7. Configure a branch in the access policy for an Outlook Anywhere client that has authenticated using NTLM.
   a) Click the **[+]** sign on the successful branch after the NTLM Auth Result action.
   An Add Item window opens.
   b) On the Assignment tab, select **SSO Credential Mapping** and click **Add Item**.
   The SSO Credential Mapping screen opens.
   c) Click **Save**.
   The properties screen closes and the policy displays.
   d) On the fallback branch after the SSO Credential Mapping action, click the **Deny** ending.
   A popup screen opens.
   e) Select **Allow** and click **Save**.
   You have completed a branch in the access policy for an Outlook Anywhere client that, having previously authenticated with NTLM, has SSO (Kerberos KCD) access on the back end.
8. Configure a branch in the access policy for an Outlook Anywhere client that uses HTTP Basic authentication.
   a) Click the **[+]** sign on the fallback branch after the NTLM Auth Result action.
   An Add Item window opens.
   b) On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
   c) Make any changes that you require to logon page properties and click **Save**.
   The properties screen closes and the policy displays.
   d) On the Successful branch after the Logon Page action, add an authentication action.
   e) On the Successful branch after the authentication action, add an SSO Credential Mapping action.
   f) On the fallback branch after SSO Credential Mapping, change the ending from Deny to Allow.

   You have completed a branch in the access policy to authenticate an Outlook Anywhere client that uses HTTP Basic authentication and provides SSO (Kerberos KCD) access for the client on the back end.
9. (Optional) On the fallback branch after the MS Exchange Client action, configure a branch for a client that is not an Outlook Anywhere client.

You could add Logon Page, authentication, and SSO Credential Mapping actions or other actions here.

10. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

You have created an access policy that checks whether the client is an Outlook Anywhere client and whether such a client has authenticated using NTLM. If so, the policy provides SSO (Kerberos KCD) access on the backend server.

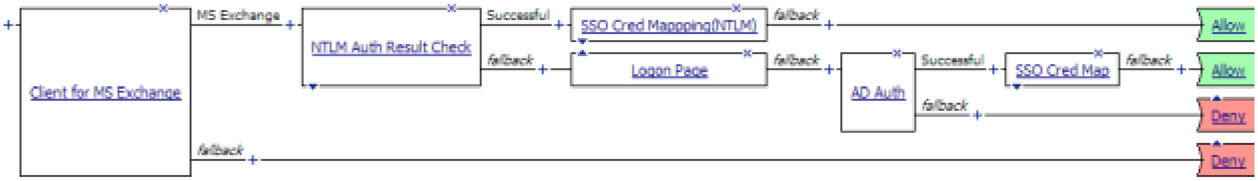


Figure 13: Example access policy with actions based on whether NTLM authentication occurred

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Maintaining a machine account

In some networks, administrators run scripts to find and delete outdated machine accounts on the domain controllers. To keep the machine account up-to-date, you can renew the password periodically.

1. On the Main tab, click **Access** > **Authentication** > **NTLM** > **Machine Account**.
   The Machine Account screen opens.
2. Click the name of a machine account.
   The properties screen opens and displays the date and time of the last update to the machine account password.
3. Click the **Renew Machine Password** button.
   The screen refreshes and displays the updated date and time.

This changes the machine account last modified time.

Updating the log level for NTLM for Exchange clients

Before you follow these steps, you must have an access profile that you configured to use for NTLM authentication of Microsoft Exchange clients. You must know the name of the log setting that is assigned to that access profile. (The default-log-setting is assigned by default, but your access profile configuration might be different.)

You can change the level of logging for NTLM authentication for Microsoft Exchange clients.

***Note:** Logging at the default level, **Notice**, is recommended.*

1. On the Main tab, click **Access** > **Overview** > **Event Logs** > **Settings**.
   A log settings table screen opens.
2. Select the check box for the log setting that you want to update and click **Edit**.
   A popup screen opens.
3. To configure settings for access system logging, select **Access System Logs** from the left pane.
   Access System Logs settings display in the right panel.
4. For the **ECA** setting, select a log level.

   ***Note:** Setting the log level to **Debug** can adversely impact system performance.*
5. Click **OK**.
   The popup screen closes.

HTTP Basic Authentication for Microsoft Exchange Clients

Overview: Configuring APM for Exchange clients that use HTTP Basic

Access Policy Manager® (APM®) requires an Exchange profile to support Microsoft Exchange clients. An Exchange profile is specified in the access profile attached to the virtual server that handles the traffic from Exchange clients.

Task summary

Configuring an Exchange profile

Creating an access profile for Exchange clients

Verifying log settings for the access profile

Configuring an access policy for Microsoft Exchange clients

Adding the access profile to the virtual server

About Exchange profiles

An Exchange profile specifies service settings for Microsoft Exchange clients. Based on the settings, Access Policy Manager® (APM®) identifies the client, authenticates the client and, when an SSO configuration is specified, provides SSO.

In an Exchange profile, you can specify settings for one or more of these Microsoft Exchange services:

- ActiveSync
- Autodiscover
- Exchange Web Service
- Offline Address Book
- Outlook Anywhere

For Microsoft Exchange clients that are configured to use NTLM, you must include an NTLM authentication configuration in the Exchange profile.

Note: *With an NTLM authentication configuration, APM supports only Kerberos SSO on the back end.*

An Exchange profile is specified in an access profile.

Configuring an Exchange profile

If any of the Microsoft Exchange clients you support authenticate using NTLM, you must first create these objects:

- A machine account
- An NTLM Auth configuration
- At least one Kerberos SSO configuration

Note: *For Access Policy Manager® (APM®) to support Kerberos SSO, a delegation account is required on Active Directory.*

You create an Exchange profile to specify how to handle traffic from Microsoft Exchange clients.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Microsoft Exchange**.
   A list of Exchange profiles displays.
2. Click **Create**.
   A Create New Exchange Profile popup screen displays general settings.
3. In the **Exchange Name** field, type a name for the Exchange profile.
4. From the **Parent Profile** list, select a profile.

   The Exchange profile inherits settings from the parent profile that you select.

   Note: *APM supplies a default Exchange profile named exchange.*
5. Repeat these steps for one or more Microsoft Exchange services:
   a) From Service Settings on the left, select an Exchange service.
      Settings for the service are displayed in the right pane.
   b) In the **URL** field, retain any default settings that are displayed or type a path to use to match the Exchange client.

      Default settings for this field are supplied in the default exchange profile.
   c) From the **Front End Authentication** list, select the type of authentication to use: **Basic**, **Basic-NTLM**, or **NTLM**.

      Only the applicable authentication types for the particular the Exchange service are included on the list.

      Note: *If you select **NTLM** or **Basic-NTLM**, you must also select a configuration from **NTLM Configuration** list on the General Settings screen.*
   d) From the **SSO Configuration** list, select an SSO configuration, if needed, for use after initial login.

      For **Basic-NTLM**and **NTLM** authentication types, only Kerberos SSO is supported.

   You configured settings for one or more Microsoft Exchange services.
6. Click **OK**.
   The screen closes.

The Exchange profile is displayed on the list.

Apply this Exchange profile by adding it to an access profile.

Creating an access profile for Exchange clients

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session. You add an Exchange profile to the access profile to specify how to handle traffic from Microsoft Exchange clients.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*
4. (Optional) In the Configurations area from the **Exchange** list, select an Exchange profile.

   Exchange profiles specify any SSO configurations for Microsoft Exchange services, such as Autodiscover, Outlook Anywhere, and so on. The configuration in the Exchange profile is used for

Microsoft Exchange clients regardless of any SSO configuration you select from the **SSO Configuration** list in this access profile.

5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.
7. To change from using the default-log-settings that APM automatically adds to the access profile, you can do this.:

   Logging occurs for a session only when a log setting is specified for the access profile.

   a) Click the name of the access profile.
      The Properties screen opens.
   b) On the menu bar, click **Logs**.
      The General Properties screen opens.
   c) In the Log Settings area, move log settings from the **Available** list to the **Selected** list.
   d) Click **Update**.

   You can configure log settings in the **Access** > **Overview** > **Event Log** > **Settings** area of the product.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy for Microsoft Exchange clients

Before you configure this access policy, you must have an AAA Active Directory server configured in Access Policy Manager®.

You configure an access policy to support Microsoft Exchange clients with login, HTTP basic authentication, and SSO.

***Note:** This access policy does not support Microsoft Exchange clients that are configured to authenticate using NTLM.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. On a policy branch, click the **(+)** icon to add an item to the policy.
   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the properties and click **Save**.
   The properties screen closes and the policy displays.
6. On the fallback branch after the previous action, click the **(+)** icon to add an item to the policy.
   A popup screen opens.
7. On the Authentication tab, select **AD Auth**.
   A properties screen displays.
8. From the **Server** list, select a server.
9. Click **Save**.
   The properties screen closes and the policy displays.
10. On the Successful branch after the previous action, click the **(+)** icon.
    A popup screen opens.
11. On the Assignment tab, select **SSO Credential Mapping** and click **Add Item**.
    A properties screen opens.
12. Click **Save**.
    The properties screen closes and the policy displays.
13. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

HTTP and HTTPS Authentication

About HTTP AAA server authentication

An HTTP AAA server directs users to an external web-based server to validate credentials. Access Policy Manager® (APM®) supports these HTTP authentication types:

- HTTP basic authentication - Directs users to a URI
- HTTP NTLM authentication - Directs users to a URI
- HTTP form-based authentication - Directs users to a form action URL and provides the specified form parameters
- HTTP custom post - Directs users to a POST URL, a submit URL, or a relative URL and provides the specified content

***Tip:** Use HTTPS instead of HTTP authentication for improved security, because HTTP authentication passes user credentials as clear text.*

Task summary for HTTP authentication

To set up this configuration, you must first configure one HTTP AAA server that supports the type of authentication that you want: HTTP Basic/NTLM, form-based, or custom post. After you configure an HTTP AAA server, you must add an HTTP Auth action to an access policy and specify the HTTP AAA server that supports the authentication type that you want to use.

Task list

Configuring an AAA server for HTTP Basic/NTLM authentication
Configuring an HTTP AAA server for form-based authentication
Configuring an HTTP AAA server for custom post authentication
Creating an access profile
Verifying log settings for the access profile
Using HTTP authentication in an access policy
Creating a virtual server

Configuring an AAA server for HTTP Basic/NTLM authentication

You configure an HTTP AAA server when you want to use Basic/NTLM authentication.

1. On the Main tab, click **Access** > **Authentication** > **HTTP**.
   The HTTP servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For **Authentication Type**, select `Basic/NTLM`.
5. In the **Start URI** field, type the complete URI that returns the logon form.

   The URI resource must respond with a challenge to a non-authenticated request.
6. Click **Finished**.

The new server displays on the list.

Configuring an HTTP AAA server for form-based authentication

You create a form-based HTTP AAA configuration to use HTTP form-based authentication from an access policy.

1. On the Main tab, click **Access** > **Authentication** > **HTTP**.
   The HTTP servers screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For **Authentication Type**, select **Form Based**.
5. (Optional) In the **Start URI** field, type a URI, for example, `http://plum.tree.lab2.sp.companynet.com/`.

   This resource must respond with a challenge to a non-authenticated request.

   Note: *This field is optional. If you type a URI in this field and you type a relative URL in the* ***Form Action*** *field, Access Policy Manager® (APM®) uses the value of the* ***Start URI*** *as the base URL; APM uses the base URL to resolve the relative URL and produce the final URL for HTTP POST.*
6. From the **Form Method** list, select either **GET** or **POST**.

   If you specify **GET**, the authentication request converts as HTTP GET.
7. In the **Form Action** field, type a URL that specifies where to process the form and perform form-based authentication. If you specified a **Start URI**, you can type a relative URL, otherwise you must type an absolute URL:

   - relative URL - When specified, form-based authentication is performed after the URL is resolved using the base URL that is specified in the **Start URI** field.
   - absolute URL -When specified, form-based authentication is performed at this URL.
8. In the **Form Parameter For User Name** and **Form Parameter For Password** fields, type the parameter name and password used by the form to which you are sending the POST request.
9. In the **Hidden Form Parameters/Values** field, type the hidden form parameters required by the authentication server logon form at your location.

   You must provide hidden form parameters and values if there are any. When present, these values are required by the authentication server logon form at your location.

   Specify a parameter name, a space, and the parameter value, if any. Start each parameter on a new line. If you use a session variable as a value, format it as shown in this example: `%{session.client.platform}`.
10. In the **Number Of Redirects To Follow** field, type how far from the landing page, in pages, the request should travel before failing.
11. For the **Successful Logon Detection Match Type** setting, select the method your authenticating server uses, and type the option definition in the **Successful Logon Detection Match Value** field.
12. Click **Finished**.
    The new server displays on the list.

Configuring an HTTP AAA server for custom post authentication

You create a custom post configuration when there is no form and when body encoding is different from form encoding. (This can happen when POST is generated by JavaScript or ActiveX.) Using a custom post, you can specify the entire post body and any non-default HTTP headers.

1. On the Main tab, click **Access** > **Authentication** > **HTTP**.

The HTTP servers screen opens.

2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For the **Authentication Type** setting, select **Custom Post**.
5. In the **Start URI** field, type in a URL resource, for example, `http://plum.tree.lab2.sp.companynet.com/`.

   If you do not specify a Start URI, Access Policy Manager® will likely detect that the absolute URI based on the Form Action parameter should be used for HTTP POST. If you specify a Start URI, Access Policy Manager uses both the Start URI and the Form Action parameters as the final URL for HTTP POST.
6. In the **Form Action** field, type the POST URL, the submit URL, or a relative URL.
7. For the **Successful Logon Detection Match Type** setting, select the method that the authenticating server uses.
8. For the **Successful Logon Detection Match Value**, type a value depending on the **Successful Logon Detection Match Type** that you selected:

   - **By Resulting Redirect URL** - Specify a URL if you selected this type.
   - **By Presence of Specific String in Cookie** - Specify a single string if you selected this type.

     Note: *With this option, when APM® receives a duplicate cookie, it adds it to the existing cookie list. As a result, multiple cookies with the same name, domain, and path can exist and can be searched.*
   - **By Presence of Cookie That Exactly Matches** - Specify the exact key fields (name, path, and domain) that are present in the HTTP response cookie if you select this type. Failure to supply the exact number of keys and the exact values for the HTTP response cookie results in a `No matching cookie found` error.

     Note: *This option supports cookie merge functionality. When APM receives a cookie that has the same name, domain, and path as an existing cookie, it merges it into the existing cookie.*
   - **By Specific String in Response** - Specify a string if you select this option.
9. In the **Number Of Redirects To Follow** field, type how far from the landing page, in pages, the request should travel before failing.
10. From the **Content Type** list, select an encoding for the HTTP custom post.

    The default setting is **XML UTF-8**.

    Note: *If you select* ***None****, you must add a header in the* ***Custom Headers*** *setting and you must apply your own encoding through an iRule.*
11. In the **Custom Body** field, specify the body for the HTTP custom post.
12. For **Custom Headers**, specify names and values for header content to insert in the HTTP custom post.
13. Click **Finished**.
    The new server displays on the list.

This creates an HTTP AAA server that provides a custom post for authentication.

To put this authentication into effect, add this AAA server to an HTTP Auth action in an access policy.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   ***Note:** A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     ***Note:** No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     ***Note:** You can edit Identity Service profile properties.*

   ***Note:** Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.

The properties screen opens.

3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Using HTTP authentication in an access policy

Before you can set up an access policy to use HTTP authentication, you must have at least one HTTP AAA server configured.

You configure an access policy with an HTTP Auth action when you want users to authenticate using one of the HTTP authentication types that Access Policy Manager® (APM®) supports: Basic, NTLM, form-based, or custom.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **HTTP Auth** and click **Add item**.
   A properties popup screen opens.
8. From the **AAA Server** list, select the AAA HTTP server you want to use for authentication.
9. (Optional) Add any other branches and actions that you need to complete the policy.
10. Click **Save**.
    The properties screen closes and the policy displays.
11. Click **Apply Access Policy** to save your configuration.

This adds an HTTP AAA authentication server to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Creating a virtual server

When creating a virtual server for an access policy, specify an IP address for a single host as the destination address.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **HTTP Profile** list, select **http**.
7. If you use server SSL for this connection, from the **SSL Profile (Server)** list, select a server SSL profile.
8. If you use client SSL for this profile, from the **SSL Profile (Client)** list, select a client SSL profile.
9. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
10. From the **Connectivity Profile** list, select a connectivity profile.

    You can select the default connectivity profile, **connectivity** if you have not defined a specific profile for the traffic that is directed to this virtual server.
11. Click **Finished**.

You have configured a host virtual server and associated an access profile with it.

Overview: Configuring HTTPS authentication

You can configure HTTP AAA authentication to use server-side SSL (HTTPS). To set up this configuration, you must first configure one HTTP AAA server that supports the type of authentication that you want to use: HTTP Basic/NTLM, form-based, or custom post.

HTTP AAA server configuration notes

Configure the HTTP AAA server so that in the **Start URI** or **Form Action** field you use:

- The http scheme (not https)
- The host name of the external HTTP server (rather than the IP address)

For example: `http://plumtree.lab2.sp.companynet.com`.

Virtual server configuration notes

Configure the virtual server to use the host name of the external HTTP server; this is the same host name as used in the HTTP AAA server configuration.

***Important:** Set the **Destination** field to use the host name of the external HTTP server. For example:* `companynet.com` *(and set the **Service Port** to HTTP).*

To ensure that SSL is used between the HTTP AAA server and the external HTTP server, the virtual server configuration includes a server SSL profile and a pool with a member that uses SSL.

DNS configuration notes

The DNS configuration on the BIG-IP® system must send traffic to the virtual server instead of the external HTTP server.

***Note:** This implementation does not explain how to configure DNS.*

Task summary

Before you start these tasks, configure an HTTP AAA server.

Creating a pool for HTTPS authentication
Creating a virtual server for HTTPS authentication
Creating an access profile
Verifying log settings for the access profile
Using HTTP authentication in an access policy
Adding the access profile to the virtual server

Creating a pool for HTTPS authentication

You create a pool (HTTPS) so that you can assign it to a virtual server (HTTP) that accepts HTTP traffic and provides server-side SSL using this pool.

1. On the Main tab, click **Local Traffic** > **Pools**.
   The Pool List screen opens.
2. Click **Create**.
   The New Pool screen opens.
3. In the **Name** field, type a unique name for the pool.
4. Scroll down to the Resources area.
5. In the **New Members Address** field, type an IP address.
6. From the **Service Port** list, select `HTTPS`.
7. Click **Add**.
8. Click **Finished**.

Creating a virtual server for HTTPS authentication

You create a virtual server that accepts HTTP traffic, encrypts it (using a server SSL profile), and passes it to an HTTPS server to provide secure communication between the BIG-IP® system and an external HTTP authentication server.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address of the external HTTP server.
5. From the **Service Port** list, select **HTTP**.

6. From the **SSL Profile (Server)** list, select a profile.

   This ensures that there is an SSL connection between the HTTP virtual server and the external HTTPS server.
7. From the **VLAN and Tunnel Traffic** list, select **Enabled on...**
8. From the **Source Address Translation** list, select **Auto Map**.
9. Scroll all the way down to the Resources area and from the **Default Pool** list, select the pool you configured previously.

   The pool must contain a member configured for HTTPS.
10. Click **Finished**.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*
4. From the **Profile Type** list, select one these options:

   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*
   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Using HTTP authentication in an access policy

Before you can set up an access policy to use HTTP authentication, you must have at least one HTTP AAA server configured.

You configure an access policy with an HTTP Auth action when you want users to authenticate using one of the HTTP authentication types that Access Policy Manager® (APM®) supports: Basic, NTLM, form-based, or custom.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

Note: Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.

A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.

7. On the Authentication tab, select **HTTP Auth** and click **Add item**.
   A properties popup screen opens.
8. From the **AAA Server** list, select the AAA HTTP server you want to use for authentication.
9. (Optional) Add any other branches and actions that you need to complete the policy.
10. Click **Save**.
    The properties screen closes and the policy displays.
11. Click **Apply Access Policy** to save your configuration.

This adds an HTTP AAA authentication server to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Local User Database

Overview: Configuring and administering a local user database

You can create multiple local user databases to provide on-box authentication, to control user access, to segment your users, and to store user information.

During access policy operation, you can read from and write to a local user database.

- You can read from a local user database to:
  - Determine whether a user is locked out of a local user database instance.
  - Check the number of failed login attempts for a user.
  - Check group membership for the user to determine which access policy branch to take.

  Note: *Groups are text strings. You create them from the Configuration utility.*
- You can write to a local user database primarily to increment or reset the number of login failures for a user. You can also update the locked out status for the user; although this option provides flexibility, use it sparingly. Normally, locked out status is set programmatically.

Task summary

Configuring a local user database instance

Adding a user to a local user database instance

Forcing change of password for a local user database instance

About backing up and restoring users

You can export user data from a local user database instance to a comma-separated values (CSV) file. The purpose is to provide you with a way to back up user data, which you can import from the CSV file.

Note: *Dynamically created users are not included in the CSV file that you export to back up user data. (You can configure the* ***Local Database*** *action to dynamically create users from an access policy.)*

About local user database synchronization across devices

When BIG-IP® systems are included in a Sync-Failover device group, configuration data is synchronized automatically or manually using ConfigSync. ConfigSync has no effect on local user databases, however.

For local user database data to be synchronized across devices, the devices must be included in a Sync-Failover device group. In a Sync-Failover device group, the active node provides the local user database data to the other nodes initially, and then provides updated data every five minutes.

Synchronization status

The date and time of the last high availability (HA) synchronization displays in the Configuration utility on the Local User List and Local Database Instance screens. If errors occur, they are logged and then available in Access Policy Manager® reports.

About updates to a local user database

Administrators can use the Configuration utility to update user data in a local database instance. A Local Database action in an access policy can be configured to update user data also. When this is the case, the act of changing user data from the Configuration utility can prevent an access policy from functioning correctly, that is, the access policy might fail to lock users out.

Depending on how a Local Database action is configured, it can modify the number of login failures and the locked out status for a given user. For a user that is not found in the local database, the Local Database action can dynamically create a user record and use it to lock the user out. To keep the user locked out, the user record must be retained for the lockout interval. (Dynamically created users are deleted automatically after a configurable period of time.)

Note: *The Local Database action can write not only to a local user database in the partition where the access policy was created, but to a local user database in the Common partition. (Usually, access to the Common partition from any other partition is read-only.)*

Configuring a local user database instance

Configure a local user database instance so you can add users and user data to it.

1. On the Main tab, select **Access** > **Authentication** > **Local User DB** > **Instances**.
   The Instances screen displays.
2. Click **Create New Instance**.
   The Create New Local User DB Instance popup screen displays.
3. In the **Name** field, type a unique name for the database instance.
4. In the **Lockout Interval (in seconds)** field, type the number of seconds to keep a user account locked.

   The default setting is 600 seconds.

   Note: *Access Policy Manager® (APM®) unlocks the user account after the interval completes; however, this does not occur immediately. The actual duration varies depending on the workload on the process that unlocks the user account.*
5. In the **Lockout Threshold** field, type the maximum number of login failures to allow.

   The default setting is 3.
6. In the **Dynamic User Remove Interval (in seconds)** field, type the number of seconds to keep a user account locked.

   The default setting is 1800 seconds.

   Note: *APM deletes the dynamic user after the interval completes; however, this does not occur immediately. The actual duration varies depending on the workload on the process that deletes the user.*
7. Click **OK**.

You have created a local user database instance.

Adding a user to a local user database instance

Before you start this procedure, a local user database instance must already exist.

Add a user to a local user database instance for authentication or for determining a branching strategy in an access policy that is based on user group membership.

***Note:** The data in a local user database is not validated against external sources.*

1. On the Main tab, select **Access** > **Authentication** > **Local User DB** > **Users**.
   The Users screen displays.
2. Click **Create New User**.
   The Create New Local User screen opens and displays User Information settings.
3. In the **User Name** field, type the user name.
4. In the **Password** and **Confirm Password** fields, type the user's password.
5. Select the **Force Password Change** check box to force the user to change password the next time they log in.

   After the user successfully changes password, this check box is cleared. You can select this check box at any time to force the user to change password at their next log in.
6. From the **Instance** list, select a local user database instance.
   You have completed the mandatory settings.
7. (Optional) Select **Personal Information** to specify **First Name**, **Last Name**, and **Email**.

   This information is not accessible to an access policy.
8. (Optional) Select **User Groups** to specify **Group Memberships** for the user.

   A group membership is a text string.

   A **Local Database** action can read groups from the database to determine branching strategy in an access policy.
9. Click **OK**.

You have created a local user in a local user database instance.

To add this user to another local user database instance, repeat this procedure and select the other instance.

Forcing change of password for a local user database instance

You can force a user to change password for the local user database instance when you need to do so.

1. On the Main tab, select **Access** > **Authentication** > **Local User DB** > **Users**.
   The Users screen displays.
2. Select a user and click **Edit**.
   The Edit Local User screen opens and displays **User Information** settings.
3. Select the **Force Password Change** check box.

   You can select this check box at any time to force the user to change a password at their next log in.
4. Click **OK**.
   The screen closes.

The user is prompted to change a password the next time they log in. After the user successfully changes a password, the **Force Password Change** check box is cleared.

Overview: Using a local user database to control authentication

You can authenticate users directly against a local user database using the Local DB Auth action from an access policy.

Furthermore, you can use the local database to count login failures by users, whether or not they are found in a user database, and locking them out for a period of time. (This is possible whether authentication occurs against an external AAA server, or a local user database.) You can accomplish these tasks by using the BIG-IP® system's Local Database actions to read and write information from an access policy.

Task summary

Authenticating users and locking them out with a local database
Unlocking a user who is locked out of a local user database instance

About locking a user out of an AAA server using a local user database

A macro, AD auth and LocalDB lockout, is available in the visual policy editor and provides a good example of using the Local Database action to lock users out of an external AAA server.

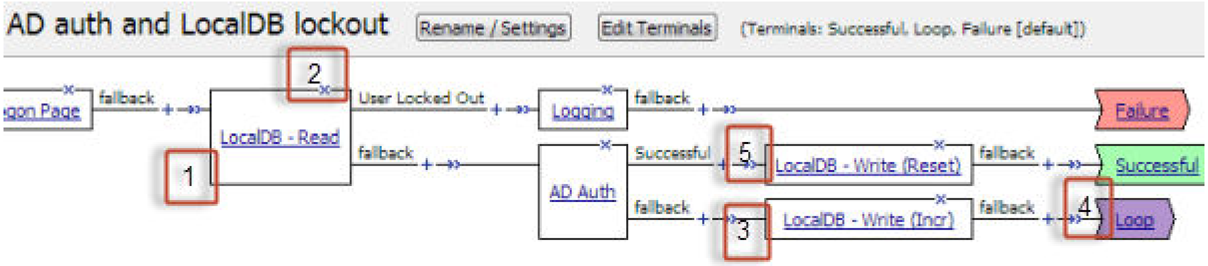


Figure 14: AD auth and LocalDB lockout macro

1. A local database action (LocalDB - Read) reads the **locked_out** database property and determines whether the user is locked out.
2. Exiting the LocalDB - Read action on the User Locked Out branch leads to a logging action and a Failure terminal.
3. If the login (AD Auth) fails, a local database write action (LocalDB - Write (Incr)) increments the number of login failures by 1.

   Note: *To keep actual logon attempts aligned with the number recorded by the LocalDB - Write (Incr) action (incrementing by 1), the* ***Max Logon Attempts Allowed*** *property in the AD Auth action is set to 1. If it was set to another number, for example 2, you would need to configure the LocalDB - Write (Incr) action to increment login failures by the same number, 2.*

4. A Loop terminal in the macro causes the macro to loop through the AD Auth and LocalDB - Write (Incr) actions until authentication succeeds or until the maximum number of logon attempts is surpassed and the macro exits through the Loop terminal.
5. If the login, AD Auth, succeeds, a Local Database write action, LocalDB - Write (Reset), resets the user's login failures to 0 (zero).

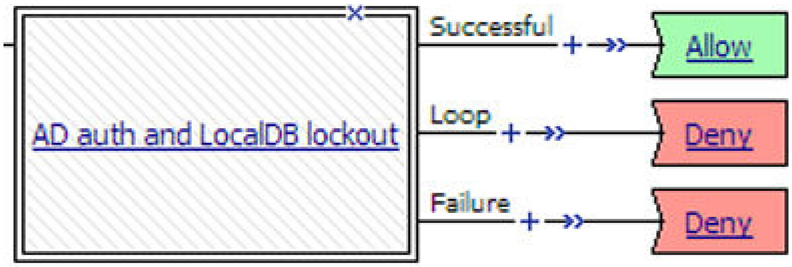


Figure 15: AD auth and LocalDB lockout macrocall in an access policy

In an access policy, three branches follow the macro:

- Successful: The user logged in successfully.

- Loop: The user failed to log in the maximum number of times and is locked out now.
- Failure: The user is locked out and does not get another chance to try to log in.

About updates to a local user database

Administrators can use the Configuration utility to update user data in a local database instance. A Local Database action in an access policy can be configured to update user data also. When this is the case, the act of changing user data from the Configuration utility can prevent an access policy from functioning correctly, that is, the access policy might fail to lock users out.

Depending on how a Local Database action is configured, it can modify the number of login failures and the locked out status for a given user. For a user that is not found in the local database, the Local Database action can dynamically create a user record and use it to lock the user out. To keep the user locked out, the user record must be retained for the lockout interval. (Dynamically created users are deleted automatically after a configurable period of time.)

Note: *The Local Database action can write not only to a local user database in the partition where the access policy was created, but to a local user database in the Common partition. (Usually, access to the Common partition from any other partition is read-only.)*

Authenticating users and locking them out with a local database

Before you start this task: create a local user database instance in Access Policy Manager®, add users to the local database instance, and then create an access profile.

Authenticate a user against a local user database when an external AAA server is not available. Read and write to a local user database when you want to track failed login attempts and lock out users that repeatedly attempt and fail to log in.

Note: *For enhanced security, F5® recommends that you place Local Database actions before and after a LocalDB Auth action to read and write user information. This enables you to track and block login attempts by any user. This process is demonstrated in the example access policy described here. (You can use this same process to lock users out of an AAA server by substituting another authentication action for the LocalDB Auth action.)*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. On a policy branch, click the **(+)** icon to add an item to the policy.

   Repeat this action from the visual policy editor whenever you want to add an item to the policy.

   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
4. On the selection screen, type `logon` in the search field, select **Logon Page** from the results, and select **Add Item**.

   A logon page should precede other actions in the access policy.

   Note: *A locked out user is not presented with a logon page, regardless of how many authentication attempts are allowed.*
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. On the fallback branch after the previous action, click the **(+)** icon to add an item to the policy.
   A popup screen opens.

7. Add a Local Database action and configure it to read the user's locked out status, and create a branch for a locked out user.
   a) Type `local` in the search field.

      Search is not case-sensitive.

      A list of matching actions displays.
   b) Select **Local Database** and click **Add Item**.
      A properties screen opens.
   c) From the **LocalDB Instance** list, select a local user database.
   d) In the **User Name** field, retain the default session variable or type another variable name or a user name.
   e) Click **Add new entry**
      A new line is added to the list of entries with the Action set to **Read** and other default settings.
   f) In the Destination column **Session Variable** field, type `session.localdb.locked_out` (or type the name of another variable).
   g) In the Source column from the **DB Property** list, select **locked_out**.
      The entry is complete. A read action reads a value from the database into a session variable.
   h) Click the Branch Rules tab.
   i) Click **Add Branch Rule**.
      A new entry with **Name** and **Expression** settings displays.
   j) In the **Name** field, replace the default name by typing a new name.

      The name appears on the branch in the policy.
   k) Click the **change** link next to the Expression setting.
      A popup screen opens.
   l) Click the **Add Expression** button.
      Settings are displayed.
   m) From the **Agent Sel.** list select **LocalDB**.
      The **Condition** list displays **LocalDB Auth Passed**. The **LocalDB Auth Passed** list displays **Passed**.
   n) From the **LocalDB Auth Passed** list, select **Locked User Out**.
      The branch rule is complete.
   o) Click **Finished**.
      The popup screen closes.
   p) Click **Save**.
      The properties screen closes and the policy displays.
8. On the fallback branch after the previous action , add a **LocalDB Auth** action and configure properties for it.

   Valid values for the properties, **LocalDB Instance** and **Max Logon Attempts Allowed**, are available from lists.

   Note: *A user that accumulates the maximum number of logon failures specified in the* ***LocalDB Auth*** *action is locked out of the local user database instance and exits the action on a Locked User Out branch. A user that is not found in the local user database exits the action to the fallback branch.*
9. On the fallback branch after the previous action, click the **(+)** icon to add an item to the policy.
   A popup screen opens.
10. Add a Local Database action to to increment and write login failures. Configure it to allow user creation.
    a) Set the **LocalDB Instance** and **User Name** fields to the same values you selected previously.
    b) From the **Allow User Creation** list, select **Yes**.

       At each subsequent login attempt, login failures increase until the user is eventually locked out.
    c) Add a new entry and configure it to read the login_failures DB property into a session variable.

d) Add a new entry and configure it to write the value of an expression that increments the number of failures into the login_failures DB property.

Here is an example of the expression:

```
expr { [mcget {session.localdb.login_failures}] + 1 }.
```

11. On the successful branch after the Local DB Auth action, add a **Local Database** action and configure it to reset login failures to 0 (zero).
    a) Set the **LocalDB Instance** and **User Name** fields to the same values you selected previously.
    b) Add a new entry and configure it to write the value of an expression that evaluates to zero into the login_failures DB property.

    Here is an example of the expression:

    ```
    expr { "0" }.
    ```
12. Click **Save**.
    The properties screen closes and the policy displays.
13. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

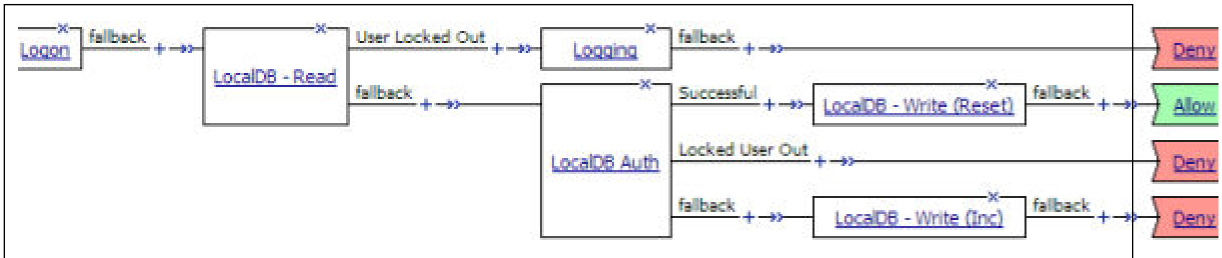


Figure 16: Sample access policy (with logging action on the User Locked Out branch)

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Unlocking a user who is locked out of a local user database instance

You can unlock a user who is locked out of a local user database instance if you do not want to wait the for lockout interval to elapse.

Note: *The lockout interval is configurable and can be different for each local user database instance.*

1. On the Main tab, select **Access** > **Authentication** > **Local User DB** > **Users**.
   The Users screen displays.
2. Select a user for whom the Locked Out column specifies **yes**.
3. Click **Unlock User**.

The account is unlocked in the local user database instance.

Overview: Branching in an access policy based on local user database groups

You can store user group membership strings in a local user database instance. You can add one or more strings for a user to the database. The strings can reflect any grouping strategy that you want to apply.

You can make user group-based branching decisions in an access policy by reading the group information for the user from the database, and creating rules for branching based on it.

Before you can perform this task, you need users and user group membership strings configured in a local user database instance. You also need an access profile.

Task summary

Creating an access policy to branch based on local DB group membership
Verifying log settings for the access profile

Creating an access policy to branch based on local DB group membership

You can use an access policy to retrieve user group membership from a local user database instance and configure branch rules to provide different actions for users in different groups.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. On a policy branch, click the **(+)** icon to add an item to the policy.

   Repeat this action from the visual policy editor whenever you want to add an item to the policy.

   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
4. Type `local` in the search field.

   Search is not case-sensitive.

   A list of matching actions is displayed.
5. Select **Local Database** and click **Add Item**.
   A properties screen displays.
6. From the **LocalDB Instance** list, select a local user database.
7. In the **User Name** field, retain the default session variable or type another variable name or a user name.
8. From the **Allow User Creation** list, retain the default value (**No**).
9. Click **Add new entry**.
   A new line is added to the list of entries.
10. Configure the entry to read the groups from the database and store them in a variable:
    a) From the Action list, select **Read**.
    b) In the **Source** column from the **DB Property** list, select **groups**.
    c) In the **Destination** column **Session Variable** field, retain the default value, session.localdb.groups or type the name of a variable.
    d) In the **Source** column from the **DB Property** list, select **groups**.
    You have configured an action that reads the user's groups into a variable.
11. Click the Branch Rules tab to edit a branch rule.
12. Click the **Add Branch Rule** button.
    New **Name** and **Expression** settings display.
13. In the **Name** field, replace the default name by typing a new name over it.

    The default name is Branch Rule *n* where *n* is a number. The name appears on the branch in the policy and so should be descriptive.
14. Click the **change** link in the Expression area.
    A popup screen opens.
15. Click the Advanced tab.

Use this tab to enter Tcl expressions.

A text input field displays.

16. Type an expression into the text input field.
    If you expect groups to include only one entry, you can type an expression similar to this one.

    ```
    expr { [mcget {session.localdb.groups}] eq "eng" }
    ```

    If you expect groups to include multiple entries, you can type an expression similar to this one

    ```
    expr { [mcget {session.localdb.groups}] contains "sales" }
    ```

17. Click **Finished**.
    The popup screen closes.
18. Add more branch rules to provide branches for different user groups.
19. Click **Save**.
    The properties screen closes and the policy displays.
20. (Optional) Add any other branches and actions that you need to complete the policy.

When the access policy runs and takes the branch with the Local Database read action, additional branching is done based on group membership.

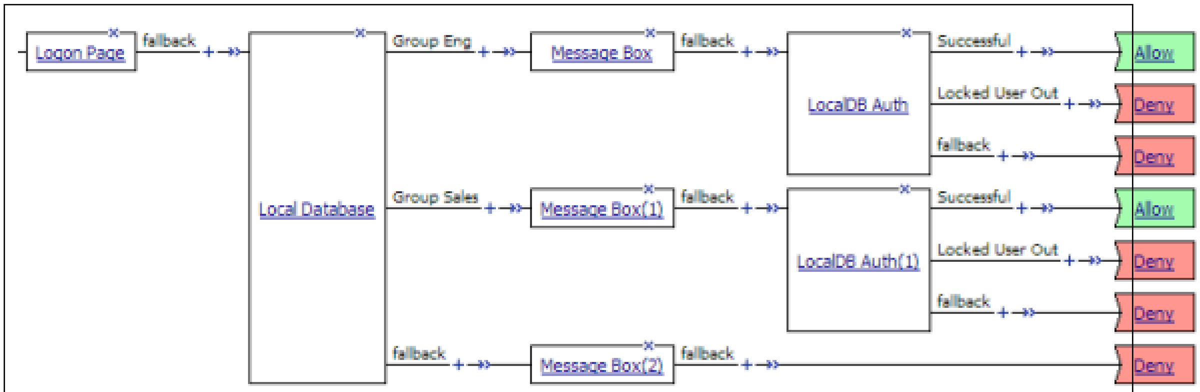


Figure 17: Sample access policy that uses local user DB groups in a branching strategy

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

Note: *Logging is disabled when the* ***Selected*** *list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

OCSP Authentication

About OCSP authentication

Access Policy Manager® (APM®) supports authenticating a client using Online Certificate Status Protocol (OCSP). *OCSP* is a mechanism used to retrieve the revocation status of an X.509 certificate by sending machine or user certificate information to a remote OCSP responder. This responder maintains up-to-date information about the certificate's revocation status. OCSP ensures that APM always obtains real-time revocation status during the certificate verification process.

Overview: Verifying machine certificate revocation status with OCSP

Access Policy Manager® supports using Online Certificate Status Protocol (OCSP) to verify the revocation status of a machine certificate.

You must have already configured the access profile to which you want to add OCSP authentication.

Task summary

Configuring an OCSP responder

Adding OCSP machine certificate verification to an access policy

Configuring an OCSP responder

Before you can specify a certificate authority file for an OCSP responder, you must import it in PEM format to the BIG-IP® system SSL certificate list.

Important: *The OCSP responder does not work with a certificate authority file that is in DER encoding format. If you've got a certificate authority file in DER format, transform it to PEM format before you import it into the BIG-IP system.*

Create an OCSP responder in Access Policy Manager® (APM®) when you want to obtain revocation status for a user or machine certificate as part of your access control strategy.

Note: *You must create one OCSP responder object in APM for each external OCSP responder from which you intend to request status.*

1. On the Main tab, click **Access** > **Authentication** > **OCSP Responder**.
   The OCSP Responder servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. In the **URL** field, type the URL used to contact the OCSP service on the responder.

   You can skip this step if you did not select the **Ignore AIA** check box and all users have certificates with the correct AIA structure. (The **Ignore AIA** option is available when you select **Advanced** from the **Configuration** list; it is disabled by default.)
5. (Optional) From the **Certificate Authority File** list, select an SSL certificate.
6. Click **Finished**.

The new server displays on the list.

You can select this OCSP responder from an OCSP Auth access policy item.

Adding OCSP machine certificate verification to an access policy

Add an OCSP Auth action to an access policy when you want to verify the revocation status of a machine certificate as part of your authentication strategy.

Important: *Before the OCSP Auth action runs, session variables must be populated with certificate data. Typically, a Machine Cert Auth action populates these variables. As an alternative, variable assignment is possible.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. Type **mach** in the search field, select **Machine Cert Auth** from the results, and click **Add Item**.

   Access Policy Manager® supports **Machine Cert Auth** for Mac and Windows-based clients.

   A Properties popup screen displays.
5. Specify values for the **Certificate Store Name**, **Certificate Store Location**, and **CA Profile** fields.
6. From the **Save Certificate in a session variable**, select **Enabled**.

   Important: *If this setting is not enabled, the OCSP Auth action cannot use the data from the X.509 certificate that the* ***Machine Cert Auth*** *action receives.*
7. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
8. Select `OCSP Auth`, and click **Add item.**
   A properties popup screen opens.
9. From the **OCSP Responder** list, select an OCSP responder.
10. From the **Certificate Type** list, select **Machine**.
11. Click **Save**.
    The properties screen closes and the policy displays.
12. Click **Apply Access Policy** to save your configuration.

Actions are added to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Overview: Verifying user certificate revocation status with OCSP

Access Policy Manager® supports using Online Certificate Status Protocol (OCSP) to verify the revocation status of a user certificate.

You must have already configured the access profile to which you want to add OCSP authentication.

Task summary

Configuring an OCSP responder

Adding OCSP user certificate verification to an access policy

Configuring a client SSL profile for OCSP

Adding client-side SSL and access profiles to a virtual server

Configuring an OCSP responder

Before you can specify a certificate authority file for an OCSP responder, you must import it in PEM format to the BIG-IP® system SSL certificate list.

Important: *The OCSP responder does not work with a certificate authority file that is in DER encoding format. If you've got a certificate authority file in DER format, transform it to PEM format before you import it into the BIG-IP system.*

Create an OCSP responder in Access Policy Manager® (APM®) when you want to obtain revocation status for a user or machine certificate as part of your access control strategy.

Note: *You must create one OCSP responder object in APM for each external OCSP responder from which you intend to request status.*

1. On the Main tab, click **Access** > **Authentication** > **OCSP Responder**.
   The OCSP Responder servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. In the **URL** field, type the URL used to contact the OCSP service on the responder.

   You can skip this step if you did not select the **Ignore AIA** check box and all users have certificates with the correct AIA structure. (The **Ignore AIA** option is available when you select **Advanced** from the **Configuration** list; it is disabled by default.)
5. (Optional) From the **Certificate Authority File** list, select an SSL certificate.
6. Click **Finished**.
   The new server displays on the list.

You can select this OCSP responder from an OCSP Auth access policy item.

Adding OCSP user certificate verification to an access policy

Add an OCSP authentication item to an access policy when you want to verify the revocation status of a user certificate as part of your authentication strategy.

Note: *Before the OCSP Auth action runs, session variables must be populated with certificate data. Typically, in an access policy either a Client Cert Inspection or On-Demand Cert Auth action receives an*

X.509 certificate from a user and stores data in session variables that the OCSP Auth action uses. As an alternative for populating session variables, variable assignment is possible.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. From the Authentication tab, select either `Client Cert Inspection` or `On-Demand Cert Auth`, and click **Add item**.

   Client Cert Inspection checks the result of an SSL handshake request that occurs at the start of an SSL session. On Demand Cert Auth performs an SSL re-handshake and checks the result. The CRLDP and OCSP Auth actions require certificate information made available by one of these policy items.
5. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
6. Select `OCSP Auth`, and click **Add item.**
   A properties popup screen opens.
7. From the **OCSP Responder** list, select an OCSP responder.
8. From the **Certificate Type** list, select **User**.
9. Click **Save**.
   The properties screen closes and the policy displays.
10. Click **Apply Access Policy** to save your configuration.

To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Configuring a client SSL profile for OCSP

To configure this client SSL profile correctly, you need to know whether the access policy (that will be paired with this SSL profile on a virtual server) includes the Client Cert Inspection agent or the On-Demand Cert Auth agent.

You need a client SSL profile to use OCSP authentication to verify a user certificate from an access policy.

1. On the Main tab, click **Local Traffic** > **Profiles** > **SSL** > **Client**.
   The Client SSL profile list screen opens.
2. Click **Create**.
   The New Client SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.

4. Select **clientssl** in the **Parent Profile** list.
5. Scroll down to the Client Authentication area.
6. Next to Client Authentication, select the **Custom** check box.
   The settings become available.
7. From the **Client Certificate** list, select the option that is applicable to the item you selected when you edited the policy.
   - Select **request** if the Client Cert Inspection agent is used in the policy.
   - Select **ignore** if the On-Demand Cert Auth agent is used.
8. From the **Trusted Certificate Authorities** list, select the Certificate Authority that issues the user certificates.
9. From the **Advertised Certificate Authorities** list, select the advertised Certificate Authority file for client certificate authentication.
10. Click **Finished**.

To put a client SSL profile into effect, you must add it to a virtual server.

Adding client-side SSL and access profiles to a virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
4. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
5. Click **Update** to save the changes.

The access policy and client-side SSL profiles are now associated with the virtual server.

OCSP session variables

When the OCSP Auth access policy item runs, it relies on information stored in session variables. Various access policy items can populate the session variables. This table lists the session variables and access policy items that can populate them.

Session variables for OCSP

Session Variable	Source	Description
`session.ssl.cert.whole`	Cert Inspection On-Demand Cert Auth Variable Assign	Provides the client certificate received from the user in PEM format. (Used for verifying the revocation status of a user certificate.)
`session.ssl.cert.certissuer`	Cert Inspection On-Demand Cert Auth Variable Assign	Provides the issuer certificate of the client certificate in PEM format. (Used for verifying the revocation status of a user certificate.)
`session.check_machinecert.last.cert.cert`	Machine Cert Auth Variable Assign	Provides the encrypted text of the machine certificate. (Used for

Session Variable	Source	Description
		verifying the revocation status of a machine certificate.)
`session.check_machinecert.last.cert.issuer.cert`	Machine Cert Auth Variable Assign	Provides the issuer certificate of the machine certificate. (Used for verifying the revocation status of a machine certificate.)

OCSP authentication troubleshooting tips

You might run into problems with OCSP authentication in some instances. Follow these tips to try to resolve any issues you might encounter.

OCSP auth and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`No AAA server associated with the agent`	Make sure that a valid OCSP responder configuration is assigned to the OCSP agent in the access policy.
`User/Issuer certificate not found for the session`	The user/issuer certificate session variables are missing. For a user certificate, make sure that either the Client Cert Inspection agent or On-Demand Cert Auth agent is configured in the access policy, or, use a variable assignment agent to create session variables. For a machine certificate, make sure that the Machine Cert Auth agent is configured or use variable assignment to create the session variables.
`Failure to connect to OCSP responder (BIO callback failure)`	Make sure that the OCSP responder is up and running and reachable from the BIG-IP® system.
`Error parsing the OCSP response (invalid response)`	Indicates that no valid basic response was found in the OCSP response. Check the configuration on the remote OCSP responder.
`Error signing OCSP request`	Make sure that the signing certificate and key are valid.
`No valid nonce found in the response`	This happens when the nonce setting is enabled on the OCSP responder configuration and the received OCSP response does not contain a valid nonce. Check the remote OCSP responder connection and setting.
`Nonce verification failed`	This happens when the nonce received in the response does not match with the nonce sent in the request. Make sure that the connection from BIG-IP system to OCSP responder is secure.
`Failure to verify response`	Make sure that the OCSP responder has a valid CA and verify other certificate settings.
`Status times invalid`	Make sure that the BIG-IP system and OCSP responder clocks are in sync.
`OCSP response - Cert with serial number 'x' has been revoked`	Indicates that the status of the user, or machine, certificate is revoked.

Possible error messages	Possible explanations and corrective actions
`Failed to add cert to OCSP request`	Indicates a failure in creating the OCSP request; either the supplied user/issuer certificates are not valid or the CertID digest configured in the OCSP responder setting is not valid.
`Failed to initialize OCSP Auth Module`	This might indicate that the certificate authority file that was imported into the BIG-IP® system is in DER encoding format. Transform the certificate authority file from DER to PEM encoding format and import it again.

CRLDP Authentication

About CRLDP configuration

Access Policy Manager® supports retrieving Certificate Revocation Lists (CRLs) from network locations (distribution points). A Certificate Revocation List Distribution Point (CRLDP) AAA server defines how to access a CRL file from a distribution point. A distribution point is either an LDAP Uniform Resource Identifier (URI), a directory path that identifies the location where the CRLs are published, or a fully qualified HTTP URL.

About AAA high availability

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established, as usual.

***Note:** Although new authentications fail if the BIG-IP® system loses connectivity to the server, existing sessions are unaffected provided that they do not attempt to re-authenticate.*

APM supports the following AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+. APM supports high availability by providing the option to create a pool of server connections when you configure the supported type of AAA server.

***Note:** If you use AAA with pools, such as RADIUS pools or Active Directory pools, APM assigns each pool member with a different number for the pool member's priority group value. APM must define each pool member with a different priority group because AAA load balancing is not used. The priority group number increases automatically with each created pool member. Alternative AAA pool configurations can be defined manually using the full flexibility of Local Traffic Manager™ (LTM®) if load balancing is desired.*

Task summary for CRLDP configuration

This task list includes all steps required to set up this configuration. If you are adding CRLDP items to an existing access policy, you do not need to create another access profile.

Task list

Configuring an AAA server for CRLDP
Creating an access profile
Verifying log settings for the access profile
Configuring an access policy that uses CRLDP authentication
Configuring a client SSL profile for CRLDP
Adding client-side SSL and access profiles to a virtual server

Configuring an AAA server for CRLDP

Create a CRLDP AAA configuration to specify how to access certificate revocation lists (CRLs).

1. On the Main tab, click **Access** > **Authentication** > **CRLDP**.
   The CRLDP Servers list screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For the **Server Connection** setting, select one of these options:
   - Select **Use Pool** to set up high availability for the AAA server.
   - Select **Direct** to set up the AAA server for standalone functionality.
   - Select **No Server** to use a fully qualified HTTP URL as the CRL location.

     Note: *The ®BIG-IP system uses the URI from the user's certificate.*

   Note: *When you select* ***No Server****, the screen updates to omit the fields that are not necessary, such as* ***Server Addresses****,* ***Server Port****, and so on.*
5. If you selected **Use Pool**, type a name in the **Server Pool Name** field.

   You create a pool of servers on this screen.
6. Provide the addresses required for your server connection:
   - If you selected **Direct**, type an IP address in the **Server Address** field.
   - If you selected **Use Pool**, for each pool member you want to add, type an IP address in the **Server Addresses** field and click **Add**.

     Note: *When you configure a pool, you have the option to type the server address in route domain format:* `IPAddress%RouteDomain`.
7. If you selected **Use Pool**, you have the option to select a **Server Pool Monitor** to track the health of the server pool.
8. If you specified **Use Pool** or **Direct** for the server connection, the **Base DN** field displays; type a CRLDP base distinguished name into it.

   This setting applies for certificates that specify the CRL distribution point in directory name (dirName) format. Access Policy Manager® uses the Base DN when the value of the X509v3 attribute, `crlDistributionPoints`, is of type `dirName`. In this case, Access Policy Manager tries to match the value of the crlDistributionPoints attribute to the Base DN value. An example of a Base DN value is `cn=lxxx,dc=f5,dc=com`.

   Note: *If the client certificate includes the distribution point extension in LDAP URI format, the IP address, Base DN, and Reverse DN settings configured on the agent are ignored; they are specific to directory-based CRLDP. All other settings are applicable to both LDAP URI and directory-based CRL DPs.*
9. Click **Finished**.
   The new server displays on the list.

An CRLDP AAA server is available for use in a CRLDP Auth agent in an access policy.

Creating an access profile

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SSO**: Select to configure matching virtual servers for Single Sign-On (SSO).

     Note: *No access policy is associated with this type of access profile*

   - **RDG-RAP**: Select to validate connections to hosts behind APM when APM acts as a gateway for RDP clients.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.
   - **System Authentication**: Select to configure administrator access to the BIG-IP® system (when using APM as a pluggable authentication module).
   - **Identity Service**: Used internally to provide identity service for a supported integration. Only APM creates this type of profile.

     Note: *You can edit Identity Service profile properties.*

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.

The properties screen opens.

3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*

   ---
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy that uses CRLDP authentication

You add CRLDP authentication to an access policy when you want to verify certificate revocation status before granting a user access.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. From the Authentication tab, select either `Client Cert Inspection` or `On-Demand Cert Auth`, and click **Add item**.

   Client Cert Inspection checks the result of an SSL handshake request that occurs at the start of an SSL session. On Demand Cert Auth performs an SSL re-handshake and checks the result. The CRLDP and OCSP Auth actions require certificate information made available by one of these policy items.
5. Click **Save.**
   The popup screen closes.
6. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
7. On the Authentication tab, select **CRLDP Auth**, then click **Add item**.
   A properties popup screen opens.
8. From the **CRLDP Server** list, select a server.
9. Click **Save**.
   The popup screen closes.
10. To grant access at the end of any branch, change the ending from **Deny** to **Allow**:
    a) Click **Deny**.

       The default branch ending is **Deny**.

   A popup screen opens.

   b) Select **Allow** and click **Save**.
   The popup screen closes. The **Allow** ending displays on the branch.

11. Click **Apply Access Policy** to save your configuration.

The access policy is complete.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Configuring a client SSL profile for CRLDP

You need a client SSL profile to use CRLDP authentication from an access policy.

1. On the Main tab, click **Local Traffic** > **Profiles** > **SSL** > **Client**.
   The Client SSL profile list screen opens.
2. Click **Create**.
   The New Client SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.
4. From the **Parent Profile** list, select **clientssl**.
5. If the access policy uses On-Demand certificate authentication, perform these substeps:
   a) From the **Configuration** list, select **Advanced**.

   Additional settings display.

   b) Select the **Custom** check box for **Configuration**.
   The settings become available.

   c) In the **Ciphers** field, type the name of a NATIVE cipher.

   The list of supported NATIVE ciphers includes these:

   - RC4-MD5
   - RC4-SHA
   - AES128-SHA
   - AES256-SHA
   - DES-CBC3-SHA
   - DES-CBC-SHA
   - EXP1024-RC4-MD5
   - EXP1024-RC4-SHA
   - EXP1024-DES-CBC-SHA
   - EXP-RC4-MD5
   - EXP-DES-CBC-SHA
   - NULL-MD5
   - NULL-SHA
6. From the **Client Certificate** list, select the option that is applicable to the item you selected when you edited the policy.
   - Select **request** if the Client Cert Inspection agent is used in the policy.
   - Select **ignore** if the On-Demand Cert Auth agent is used.
7. From the **Trusted Certificate Authorities** list, select the Certificate Authority that issues the user certificates.
8. (Optional) From the **Advertised Certificate Authorities** list, select the Certificate Authority that issues the user certificates.

9. Click **Finished**.

A new client SSL profile is available.

Note: CRLDP authentication does not verify a certificate revocation list if one is selected in the client SSL profile. CRLDP authentication verifies the certificate revocation list (CRL) at a distribution point defined in the CRLDP AAA server.

Adding client-side SSL and access profiles to a virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
4. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
5. Click **Update** to save the changes.

The access policy and client-side SSL profiles are now associated with the virtual server.

Testing AAA high availability for supported authentication servers

To effectively test that high availability works for your authentication servers, you should have two servers that are accessible, where you can remove one of them from the network.

Note: *High availability is supported for these authentication server types only: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.*

If you configured a supported authentication server type to use a pool of connection servers, you can test the configuration using these steps.

1. Begin a `tcpdump` on the Access Policy Manager®, using a protocol analyzer, and scanning for packets destined for the specific port for your authentication server.
2. Log in to the virtual server with both servers active.
3. Using the `tcpdump` records, verify that the requests are being sent to the higher priority server.
4. Log out of the virtual server.
5. Disable the higher-priority server.
6. Log in to the virtual server again.
7. Verify that the request is being sent to the other server.
8. Log out again, re-enabling the server, and try one more time to verify that the new requests are being sent to the high priority server.

Example access policy for CRLDP authentication

This is an example of an access policy with all the associated elements needed to retrieve CRLs using CRLDP. Notice that you must add either the Client Cert Inspection agent or On-Demand Cert Auth agent before the CRLDP object in your access policy. One of those agents is required in order to receive the X. 509 certificate from the user. This is also important because both agents store the user information, as well as the issuer certificates, in the session variables. This allows the CRDLP Auth agent to check the revocation status of the user's certificate.

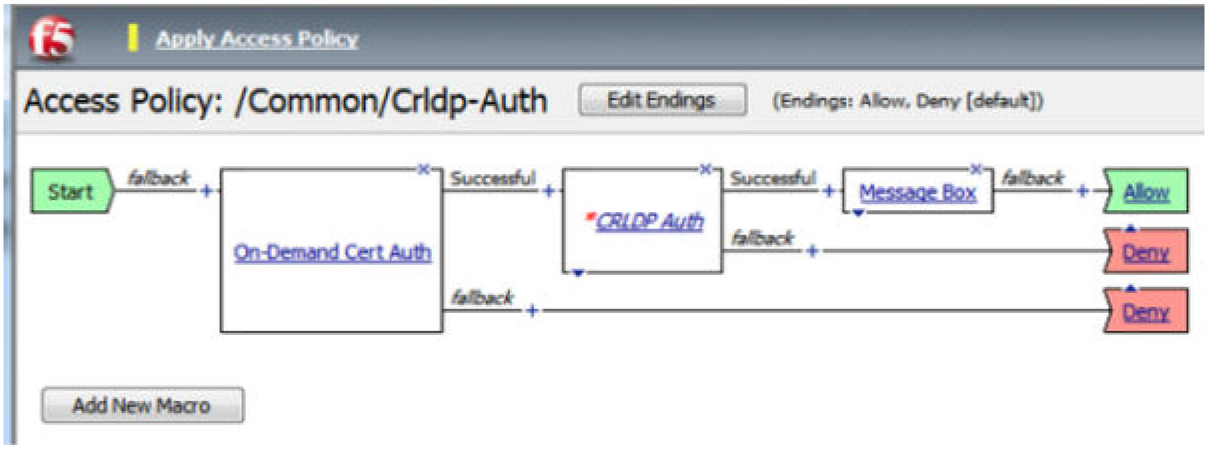


Figure 18: How CRLDP works

CRLDP session variables

When the CRLDP Auth access policy item runs, it populates session variables which are then available for use in access policy rules. The table lists the session variables for the CRLDP access policy item and for the certificate item used in the access policy.

Session variables for CRLDP

Session Variable	Description
`session.ldap.ssl.cert.whole`	Provides the client certificate received from the user in PEM format.
`session.ssl.cert.certissuer`	Provides the issuer certificate of the client certificate in PEM format.
`session.crldp.last.result`	Sets the result of the CRLDP authentication. The available values are: • 0: Failed • 1: Passed
`session.crldp.last.status`	Sets the status of the authentication to Failed.

CRLDP authentication troubleshooting tips

You might run into problems with CRLDP authentication in some instances. Follow these tips to try to resolve any issues you might encounter.

CRLDP auth and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`No AAA server associated with the agent`	Make sure that a valid CRLDP responder configuration is assigned to the CRLDP agent in the access policy.
`User/Issuer certificate not found for the session`	The user/issuer certificate session variables are missing. Make sure that either the Client Cert Inspection agent or On-Demand Cert Auth agent is configured in the access policy (or use a variable assignment agent to create them).
`Failure to connect to CRLDP server`	Make sure that the CRLDP server is up and running and reachable from the BIG-IP® system.
`No LDAP URL found in the DP list`	Indicates that no valid CRL DP is configured on the LDAP server. Make sure that the LDAP server used in the CRLDP server configuration has valid CRL DPs configured.
`CRLDP response - Cert with serial number 'x' has been revoked`	Indicates that the status of the user certificate is revoked.

On-Demand Certificate Authentication

Overview: Requesting and validating an SSL certificate on demand

Typically, when a client makes an HTTPS request, an SSL handshake request occurs at the start of an SSL session. You can configure a client SSL profile to skip the initial SSL handshake and add the On-Demand certificate authentication agent to the access policy to re-negotiate the SSL connection later. Access Policy Manager® can perform the certificate request and validation task that is normally performed by the target server, on demand.

Use the agent when you want to request and validate a certificate only after a user has already completed some other steps (logged on, gone through an authentication process, or anything else you require). Wherever you place the On-Demand authentication action in your access policy, it performs an SSL re-handshake.

You might want to use this agent, for example, if all employees must gain access to the network before only a few employees can gain access to servers with sensitive information.

Exchanging SSL certificates

Before you can use On-Demand certificate authentication successfully, you must exchange certificates between clients and the BIG-IP® system.

The client needs a valid certificate with which to respond to a certificate request. The BIG-IP system includes a self-signed certificate that you can export and install on the client. As an alternative to the self-signed certificate, you can import a certificate and corresponding key (issued by your organization CA) into the BIG-IP system and install that on the client.

The BIG-IP systems needs the client root certificate installed on it. Exporting and importing SSL certificates is done in the System File Management area of the product.

Task summary

Creating a custom Client SSL profile
Adding On-Demand certificate authentication to an access policy
Verifying log settings for the access profile
Adding client-side SSL and access profiles to a virtual server

Creating a custom Client SSL profile

1. On the Main tab, click **Local Traffic** > **Profiles** > **SSL** > **Client**.
   The Client SSL profile list screen opens.
2. Click **Create**.
   The New Client SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.
4. Select **clientssl** in the **Parent Profile** list.
5. Scroll down to the Client Authentication area.
6. Next to Client Authentication, select the **Custom** check box.
   The settings become available.
7. For the **Client Certificate** setting, select **ignore**.

   When ignore is selected, the BIG-IP® system skips the initial SSL handshake.

8. For the **Trusted Certificate Authorities** setting, select a trusted certificate authority.
9. Click **Finished**.

Adding On-Demand certificate authentication to an access policy

To successfully pass the On-Demand certificate authentication, the client browser must have a valid SSL certificate for the BIG-IP® system.

***Note:** The client browser might stop responding if the client fails to provide a certificate. We strongly recommend that you add a Decision Box action in which you ask the user whether a valid certificate is installed and provide an option to not proceed to the On-Demand Cert Auth action when a valid certificate is not installed.*

Add an On-Demand Cert Auth agent to an access policy to request and validate an SSL certificate anywhere in the session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. Select the Authentication tab.
   The tab displays a list of authentication actions.
5. Select **On-Demand Cert Auth** and click **Add Item**.
   A properties screen opens.
6. From the **Auth Mode** list, select one of these:
   - **Request** This is the default mode.
   - **Required** For an iPod or an iPhone, you must select this mode. (You can select this mode for other clients as well.)

     ***Note:** To pass a certificate check using Safari, you will be asked to select the certificate multiple times. This is expected behavior.*
7. Click **Save**.
   The properties screen closes and the policy displays.
8. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

The On-Demand Cert Auth action is included and applied to the access policy.

To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding client-side SSL and access profiles to a virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
4. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
5. Click **Update** to save the changes.

The access policy and client-side SSL profiles are now associated with the virtual server.

Client Certificate Inspection

About client certificate inspection

The Client Cert Inspection access policy item checks the result of the SSL handshake that occurs at the start of a session. It does not, however, negotiate an SSL session. It relies on settings in a client SSL profile that is added to the virtual server. The Client Cert Inspection item can provide the result of the SSL handshake, including certificate revocation status when the client SSL profile specifies a certificate revocation list (CRL).

Task summary for client certificate inspection

To complete this configuration, you need an access profile and a virtual server configured. Checking the validity of a client certificate is very likely to be one of many items you add to an access policy.

Task list

Creating a client SSL profile for certificate inspection

Configuring an access policy to confirm client certificate validity

Verifying log settings for the access profile

Creating a client SSL profile for certificate inspection

The BIG-IP® system supplies a default certificate and a `ca-bundle.crt` file that includes all well-known public certificate authority (CA) certificates for client-side processing. Before you create a client SSL profile, you might want to configure a trusted certificate to use for client-side processing. To verify certificate revocation status, you must have obtained a certificate revocation list (CRL) and imported it to the SSL Certificate List.

You create a custom client SSL profile to request an SSL certificate from the client at the start of the session. This enables a Client Cert Inspection item in an access policy to check whether a valid certificate was presented.

1. On the Main tab, click **Local Traffic** > **Profiles** > **SSL** > **Client**.
   The Client SSL profile list screen opens.
2. Click **Create**.
   The New Server SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.
4. From the **Parent Profile** list, select **clientssl**.
5. Scroll down to the Client Authentication area.
6. Next to Client Authentication, select the **Custom** check box.
   The settings become available.
7. From the **Client Certificate** list, select **request**.

   Alternatively, select **require**; however, if you do, the user must provide a valid client certificate or the connection is not allowed.
8. (Optional) If you imported a CRL, select it from the **Certificate Revocation List (CRL)** list.

   If you are using this client SSL profile in conjunction with an access policy that performs OCSP Responder authentication or CRLDP authentication, do not select a CRL.

9. Click **Finished**.

To put this client SSL profile into effect, select it in a virtual server that is configured to accept HTTPS traffic.

Configuring an access policy to confirm client certificate validity

Add a client certificate inspection item to an access policy when you want to check whether the client presented a valid certificate at the start of the session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. In the search field type `client`, then select `Client Cert Inspection` from the results list, and click **Add item**.
   A popup Properties screen displays.
5. Click **Save**.
   The properties screen closes and the policy displays.
6. Complete the policy:
   a) Add any additional policy items you require.
   b) Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.
7. Click **Apply Access Policy** to save your configuration.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

***Note:** Logging is disabled when the **Selected** list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

One-Time Password Authentication

Overview: Providing a one-time password using email

Access Policy Manager® supplies an OTP Generate access policy item that generates a one-time time-sensitive password and an OTP Verify item that verifies that a user entered the correct password before that password expired. In between the two actions, you must configure an action that delivers the one-time password to the user. To send the password in an email message, use the Email access policy item. You must have an external SMTP server and you must create an SMTP server configuration for it on the BIG-IP® system.

Related access policy macro

A macro template to configure OTP over email is available for use in an access policy. Look at the macro, AD auth query OTP by email and resources, from the visual policy editor to determine whether to use it to help you configure the access policy more quickly.

Task summary

Creating an SMTP server configuration
Creating an access policy to send an OTP using email
Verifying log settings for the access profile

Creating an SMTP server configuration

You specify the SMTP server configuration so that you can send emails through an SMTP server.

1. On the Main tab, click **System** > **Configuration** > **Device** > **SMTP**.
2. Click the **Create** button.
   The New SMTP Configuration screen opens.
3. In the **Name** field, type a name for the SMTP server that you are creating.
4. In the **SMTP Server Host Name** field, type the fully qualified domain name for the SMTP server host.
5. In the **SMTP Server Port Number** field, type a port number.

   For no encryption or TLS encryption, the default is `25`. For SSL encryption, the default is `465`.
6. In the **Local Host Name** field, type the host name used in the SMTP headers in the form of a fully qualified domain name.

   This host name is not the same as the BIG-IP® system's host name.
7. In the **From Address** field, type the email address that you want displayed as the reply-to address for the email.
8. From the **Encrypted Connection** list, select the encryption level required for the SMTP server.
9. To require that the SMTP server validates users before allowing them to send email, select the **Use Authentication** check box, and type the user name and password required to validate the user.
10. Click the **Finish** button.

You can now configure the system to use this SMTP server to send emails. For the SMTP mailer to work, you must make sure the SMTP server is on the DNS lookup server list, and configure the DNS server on the BIG-IP® system.

Creating an access policy to send an OTP using email

Before you start this task, configure an access profile.

Create an access policy like this when you need to generate and send a one-time password over email.

Note: *Look at the macro, AD query auth OTP by email and resources, to determine whether to use it to configure an access policy similar to this one.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Add actions to authenticate the user and find an email address and a mobile phone number.
   a) Click the **(+)** icon anywhere in your access profile to add a new action item.
   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
   b) On the Authentication tab, select **AD Auth** and click **Add Item**.
   A popup properties screen displays.
   c) From the **Server** list, select a server and click **Save**.
   The properties screen closes.
   d) On the Successful branch after the previous action, click the **(+)** icon.
   An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   e) On the Authentication tab, select **AD Query** and click **Add Item**.

   An AD Query is only one way to find the email address for a user. If users normally log on to your system with an email address as their username, you can get the email address using a Logon Page action.

   A popup properties screen displays.
   f) From the **Server** list, select a server.
   g) Click **Add new entry**.
   An empty entry displays under Required Attributes (optional).
   h) Type **mobile** into the **Required Attributes (optional)** field

   After the query, the session.ad.last.attr.mobile variable holds the value.
   i) Click **Add new entry**.
   An empty entry displays under Required Attributes (optional).
   j) Type **mail** into the **Required Attributes (optional)** field

   After the query, the session.ad.last.attr.mail variable holds the value.
   k) Click **Save**.
   The properties screen closes.
4. Generate a one-time password.
   a) On the Successful branch after the previous action, click the **(+)** icon.
   An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   b) On the Authentication tab, select **OTP Generate** and click **Add Item**.
   c) Click **Save**.
   The properties screen closes and the policy displays.
5. Send the OTP to the user through the Email agent.
   a) On the Successful branch after the previous action, click the **(+)** icon.

An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.

b) On the General Purpose tab, select **Email** and click **Add Item**.
c) From the **SMTP Configuration** list, select a configuration.

   The configuration specifies an external SMTP server to send the email.
d) In the **From** field, type an email address on the system.
e) In the **To** field, type an email address, a session variable, or a session variable and a string.

   For example, type `%{session.ad.last.attr.mobile}`*`@providerservice.com`* where providerservice.com is supplied by a mobile phone provider.
f) Type a subject in the **Subject** field.
g) In the **Message** field, type the one-time password and anything else the user should know.
   One Time Passcode: %{session.otp.assigned.val} Expires after use or in %{session.otp.assigned.ttl} seconds
h) Click **Save**.
   The properties screen closes and the policy displays.

6. Add a Logon Page action that requests the one-time password only.
   a) On the Successful branch after the previous action, click the **(+)** icon.
      An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   b) On the Logon tab, select **Logon Page** and click the **Add Item** button.
      The Logon Page Agent properties screen opens.
   c) From the Logon Page Agent area, on line 1 select **none** from the Type column to remove the user name input field from the logon page; do not change line 2 (password).
   d) From the Customization area in **Logon Page Input Field # 2**, type a prompt for the field.

      For example, type One-Time Passcode.
   e) Click **Save**.
      The properties screen closes and the policy displays.
7. Verify the one-time password.
   a) On the Successful branch after the previous action, click the **(+)** icon.
      An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   b) On the Authentication tab, select **OTP Verify** and click **Add Item**.
   c) Click **Save**.
      The properties screen closes and the policy displays.
8. (Optional) Add any other branches and actions that you need to complete the policy.
9. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
10. At the top of the window, click the **Apply Access Policy** link to apply and activate your changes to this access policy.
11. Click the **Close** button to close the visual policy editor.

You have an access policy that provides a user with a one-time time-based password over SMTP.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

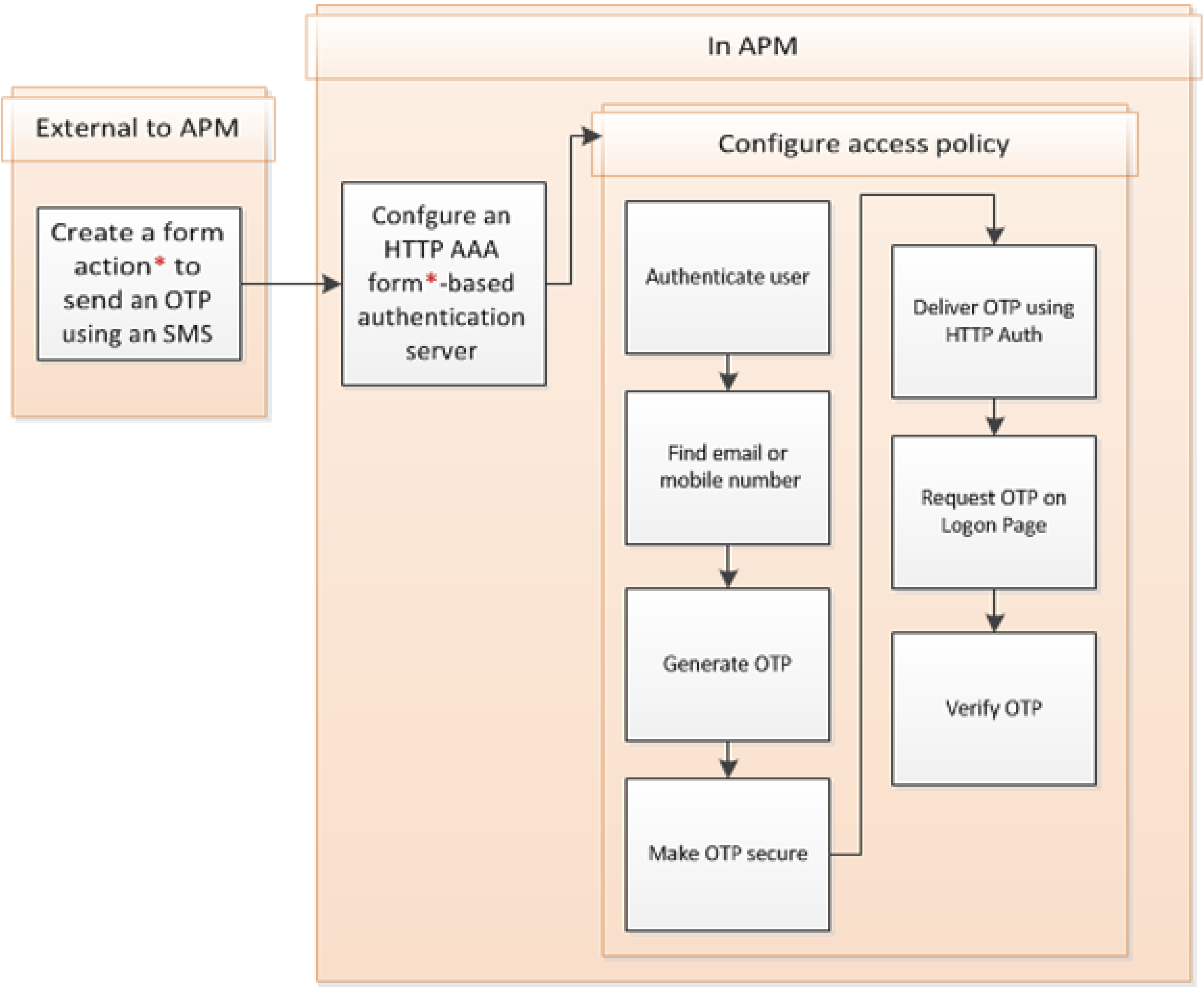
   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Overview: Providing a one-time password using an external SMS

Access Policy Manager® supplies an OTP Generate action that generates a one-time time-sensitive password and an OTP Verify action that verifies that a user entered the correct password before it expired. In between the two actions, you must configure an action that delivers the one-time password to the user. To send the password in a text message, you can use a form-based HTTP authentication agent (if you do not want to use an Email agent). You pass the one-time password in hidden parameters to a form action. You must create a form action that sends the OTP using an external SMS.

Configuration process



In APM
External to APM
Create a form action* to send an OTP using an SMS
Confgure an HTTP AAA form*-based authentication server
Configure access policy
Authenticate user
Find email or mobile number
Generate OTP
Make OTP secure
Deliver OTP using HTTP Auth
Request OTP on Logon Page
Verify OTP

Figure 19: Creating a configuration to send an OTP over SMS using HTTP authentication

Related access policy macro

A macro template to configure an OTP and use the HTTP Auth agent to deliver it is available for use in an access policy. Look at the macro, AD query auth OTP by HTTP and resources, from the visual policy editor to determine whether to use it to help you configure the access policy more quickly.

Task summary

Configuring HTTP form-based authentication to deliver a one-time password
Creating an access policy to send an OTP using an SMS
Verifying log settings for the access profile

Configuring HTTP form-based authentication to deliver a one-time password

Configure an AAA HTTP server to use a form action that you configured previously to send a one-time password through an external SMS.

1. On the Main tab, click **Access** > **Authentication** > **HTTP**.
   The HTTP servers screen opens.
2. Click **Create**.
   The New Server properties screen opens.

3. In the **Name** field, type a unique name for the authentication server.
4. From the Configuration area, select `Form Based` for the **Authentication Type**.
5. Let the **Form Method** remain at the default setting, **POST**.
6. In the **Form Action** field, type the complete destination URL to process the form.

   Specify a URL for a form action that you created to send a user a one-time password using an SMS.
7. In the **Hidden Form Parameters/Values** field, type parameters and values for the one-time password, the phone number, and any other values that the form action requires.

   Here is an example.

```
      otp_http_mobile "%{session.ad.last.attr.mobile}"
      otp_http_email "%{session.ad.last.attr.mail}"
      otp_http_body "One Time Passcode: %{session.otp.assigned.val} Expires after use or in
%{session.otp.assigned.ttl} seconds"
```

8. From the **Successful Logon Detection Match Type** list, select the method that the authenticating server uses.
9. In the **Successful Logon Detection Match Value** field, type the value that denotes successful logon.

   Type a cookie name, a URL, or a string, depending on the successful logon detection match type you selected.
10. Click **Finished**.
    The new server displays on the list.

An HTTP server for form-based authentication with a one-time password is ready for use.

Creating an access policy to send an OTP using an SMS

Before you start this task, configure an access profile and configure a form action that uses an external SMS to send the one-time password.

Create an access policy like this when you need to generate and send a one-time password as a text message and you do not want to send it using email.

***Note:** The macro, AD auth query OTP by HTTP and resources, is available from the visual policy editor and might be useful to configure an access policy similar to this one.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Add actions to authenticate the user and find a mobile phone number.
   a) Click the **(+)** icon anywhere in your access profile to add a new action item.
   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
   b) From the Authentication tab, select **AD Auth** and click **Add Item**.
   A pop-up properties screen displays.
   c) From the **Server** list, select a server and click **Save**.
   The properties screen closes.
   d) On the Successful branch after the previous action, click the **(+)** icon.
   An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   e) On the Authentication tab, select **AD Query** and click **Add Item**.
   A pop-up properties screen displays.

f) From the **Server** list, select a server.

g) Click **Add new entry**.
An empty entry displays under Required Attributes (optional).

h) Type **mobile** into the **Required Attributes (optional)** field

i) Click **Save**.
The properties screen closes.

4. Generate a one-time password.

a) On the Successful branch after the previous action, click the **(+)** icon.
An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.

b) From the Authentication tab, select **OTP Generate** and click **Add Item**.

c) Click **Save**.
The properties screen closes and the policy displays.

5. Make the OTP secure.

a) On the Successful branch after the previous action, click the **(+)** icon.
An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.

b) From the Assignment tab, select **Variable Assign** and click **Add Item**.
A properties screen opens.

c) Click **Add new entry**.
An **Empty** entry displays.

d) Click the **change** link in the new entry.
A popup screen opens.

e) From the **Unsecure** list, select **Secure**.

f) In the Custom Variable text box, type `session.user.otp.pwd`.

g) In the Custom Expression text box, type `expr { [mcget {session.user.otp.pw}]}`.

h) Click **Finished**.

The popup screen closes.

6. Send the OTP through the HTTP Auth agent.

a) On the Successful branch after the previous action, click the **(+)** icon.
An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.

b) From the Authentication tab, select **HTTP Auth** and click **Add Item**.

c) From the AAA server list, select the HTTP form-based server that you configured previously.

d) Click **Save**.
The properties screen closes and the policy displays.

7. Add a Logon Page action that requests only the one-time password.

a) On the Successful branch after the previous action, click the **(+)** icon.
An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.

b) From the Logon Page tab, select **Logon Page** and click **Add Item**.
A pop-up properties screen displays.

c) From the Logon Page Agent area, on line 1 select **password** from the Type column and change the post and session variable names.

The variable name password is acceptable.

d) From the Customization area in **Logon Page Input Field # 1**, type a prompt for the field.

For example, type One-Time Passcode.

e) Click **Save**.
The properties screen closes and the policy displays.

8. Verify the one-time password.

   a) On the Successful branch after the previous action, click the **(+)** icon.
      An Add Item screen opens, listing predefined actions that are grouped on tabs such as General Purpose, Authentication, and so on.
   b) From the Authentication tab, select **OTP Verify** and click **Add Item**.
   c) Click **Save**.
      The properties screen closes and the policy displays.

9. (Optional) Add any other branches and actions that you need to complete the policy.
10. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
11. At the top of the window, click the **Apply Access Policy** link to apply and activate your changes to this access policy.
12. Click the **Close** button to close the visual policy editor.

You have an access policy that uses HTTP authentication to provide a user with a one-time time-based password over SMS.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

TACACS+ Authentication and Accounting

About TACACS+ authentication and accounting

Access Policy Manager® (APM®) supports authenticating and authorizing the client against Terminal Access Controller Access Control System (TACACS+) servers. *TACACS+* is a mechanism used to encrypt the entire body of the authentication packet. If you use TACACS+ authentication, user credentials are authenticated on a remote TACACS+ server. If you use the TACACS+ Accounting feature, the accounting service sends `start` and `stop` accounting records to the remote server.

APM supports TACACS+ authentication with the TACACS+ Auth access policy item and supports TACACS+ accounting with the TACACS+ Acct access policy item.

Important: *APM must include a TACACS+ server configuration for every TACACS+ server that exists.*

About AAA high availability

Using AAA high availability with Access Policy Manager® (APM®), you can configure multiple authentication servers to process requests, so that if one authentication server goes down or loses connectivity, the others can resume authentication requests, and new sessions can be established, as usual.

Note: *Although new authentications fail if the BIG-IP® system loses connectivity to the server, existing sessions are unaffected provided that they do not attempt to re-authenticate.*

APM supports the following AAA servers for high availability: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+. APM supports high availability by providing the option to create a pool of server connections when you configure the supported type of AAA server.

Note: *If you use AAA with pools, such as RADIUS pools or Active Directory pools, APM assigns each pool member with a different number for the pool member's priority group value. APM must define each pool member with a different priority group because AAA load balancing is not used. The priority group number increases automatically with each created pool member. Alternative AAA pool configurations can be defined manually using the full flexibility of Local Traffic Manager™ (LTM®) if load balancing is desired.*

Task summary for TACACS+ authentication and accounting

This task list includes all steps required to set up this configuration. If you are adding TACACS+ authentication or accounting to an existing access policy, you do not need to create another access profile and the access policy might already include a logon page.

Task list

Configuring a TACACS+ AAA server for authentication and authorization

Using TACACS+ authentication in an access policy

Verifying log settings for the access profile

Configuring a TACACS+ AAA server for authentication and authorization

1. On the Main tab, click **Access** > **Authentication** > **TACACS+**.
   The TACACS+ Servers list screen opens.
2. Click **Create**.
   The New Server properties screen opens.
3. In the **Name** field, type a unique name for the authentication server.
4. For the **Server Connection** setting, select one of these options:
   - Select **Use Pool** to set up high availability for the AAA server.
   - Select **Direct** to set up the AAA server for standalone functionality.
5. If you selected **Use Pool**, type a name in the **Server Pool Name** field.

   You create a pool of servers on this screen.
6. Provide the addresses required for your server connection:
   - If you selected **Direct**, type an IP address in the **Server Address** field.
   - If you selected **Use Pool**, for each pool member you want to add, type an IP address in the **Server Addresses** field and click **Add**.

   Note: *When you configure a pool, you have the option to type the server address in route domain format:* `IPAddress%RouteDomain`.
7. If you selected **Use Pool**, you have the option to select a **Server Pool Monitor** to track the health of the server pool.
8. In the **Service Port** field, type a TACACS+ service port or select one from the list. The default is `49`.
9. In the **Secret** field, type a secret key to use to encrypt and decrypt packets sent or received from the server, and then re-type the secret key in the **Confirm Secret** field.
10. For the **Service** setting, select the name of the service for the user who is being authenticated to use.

    Identifying the service enables the TACACS+ server to behave differently for different types of authentication requests.
11. Click **Finished**.
    The new server displays on the list.

Using TACACS+ authentication in an access policy

You configure an access policy with a TACACS+ Auth action to provide TACACS+ authentication as an authentication option for users trying to gain access.

1. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
2. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
3. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
4. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.

5. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
6. Select `TACACS+ Auth`, and click **Add item.**
   A properties popup screen opens.
7. From the **AAA Server** list, select the TACACS+ server to use for authentication.
8. (Optional) Add any other branches and actions that you need to complete the policy.
9. Click **Save**.
   The properties screen closes and the policy displays.
10. Click **Apply Access Policy** to save your configuration.

This creates an access policy that presents a user with a logon page, and then uses the input credentials to authenticate the user with an external TACACS+ server specified in the TACACS+ AAA server that you select.

To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

---

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*

   ---
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Testing AAA high availability for supported authentication servers

To effectively test that high availability works for your authentication servers, you should have two servers that are accessible, where you can remove one of them from the network.

***Note:** High availability is supported for these authentication server types only: RADIUS, Active Directory, LDAP, CRLDP, and TACACS+.*

If you configured a supported authentication server type to use a pool of connection servers, you can test the configuration using these steps.

1. Begin a `tcpdump` on the Access Policy Manager®, using a protocol analyzer, and scanning for packets destined for the specific port for your authentication server.
2. Log in to the virtual server with both servers active.
3. Using the `tcpdump` records, verify that the requests are being sent to the higher priority server.
4. Log out of the virtual server.
5. Disable the higher-priority server.
6. Log in to the virtual server again.
7. Verify that the request is being sent to the other server.
8. Log out again, re-enabling the server, and try one more time to verify that the new requests are being sent to the high priority server.

Example access policy for TACACS+ authentication and accounting

This is an example of an access policy with all the associated elements needed to authenticate and authorize users with TACACS+ authentication. Note that the server used for authentication can be different from the server used for TACACS+ accounting service.

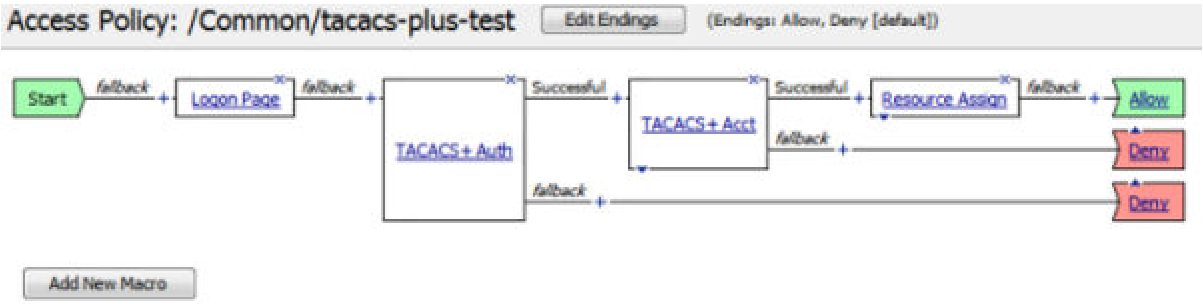


Figure 20: How TACACS Plus works

TACACS+ session variables for access policy rules

When the TACACS+ Auth (or TACACS+ Acct) access policy item runs, it populates session variables which are then available for use in access policy rules. The tables list the session variables for the TACACS+ access policy items and for a logon access policy item.

Session variables for TACACS+

Session Variable	Description
`session.tacasplus.last.acct.start_date; session.tacasplus.last.acct.start_time`	Provides TACACS+ accounting start time and date set by the accounting agent.
`session.tacacsplus.last.acctresult`	Allows the accounting agent to set the available values to either of the following values: • 0: Failed • 1: Succeeds
`session.tacacsplus.last.errmsgs`	Contains the error message string when the TACACS+ authentication or accounting fails.
`session.tacacsplus.last.result`	Sets to 1 when authentication succeeds, or 0 when it fails.

Common session variables

Session Variable	Description
`session.logon.last.username`	Provides user credentials. The `username` string is stored after encrypting, using the system's client key.
`session.logon.last.password`	Provides user credentials. The `password` string is stored after encrypting, using the system's client key.

TACACS+ authentication troubleshooting tips

You might run into problems with TACACS+ authentication in some instances. Follow these tips to try to resolve any issues you might encounter.

TACACS+ auth and query troubleshooting

Possible error messages	Possible explanations and corrective actions
`No AAA server associated with the agent`	Make sure that a valid TACACS+ server configuration is assigned to the agent (TACACS+ Auth or TACACS+ Acct) used in the access policy.
`Failure to connect to TACACS+ server`	Make sure that the TACACS+ server is up and running and reachable from the BIG-IP® system.
`Login incorrect`	Supplied user credentials are not valid.
`Invalid reply content, incorrect key`	Make sure that the shared encryption key configured on the TACACS+ server configuration matches with the key on the remote TACACS+ server.
`Invalid AUTHEN/START packet from server`	Indicates either the wrong keys or that the authentication action (LOGIN) is not supported on the server.
`Unacceptable authen method`	Indicates that the TACACS+ server does not support the authentication. Check the settings on the server.

Possible error messages	Possible explanations and corrective actions
`Unexpected failure return/legal status value from authentication function/Permission error`	Caused by internal errors on the remote TACACS+ server. Check the logs on the remote TACACS+ server and also the configuration.

APM ActiveSync Limit

Overview: Supporting larger email attachments for ActiveSync

By default Access Policy Manager® (APM®) supports a POST body of up to 64 KB for ActiveSync. If an email body exceeds that limit, APM writes a message, `ERR_NOT_SUPPORTED`, to the `var/log/apm` log file. APM can be configured to support a POST body of up to 25 MB.

Increasing the APM limit for ActiveSync email POST body

You can specify the supported size for the ActiveSync email POST body using a database variable.

***Note:** The maximum supported size is 25 MB.*

1. Log on to the BIG-IP® system command line and type `tmsh`.
2. Type this command sequence `sys db`.
3. To specify the amount of disk space allocated for hosted content:
   a) Type this command sequence `modify tmm.access.maxrequestbodysize value`.
   This prompt displays. `Values: [enter integer value min:64000 max:25000000]`
   b) Type a value and press Enter.

AAA High Availability and Upgrade

Upgrading an Access Policy Manager high availability failover pair

To ensure that upgrading a failover pair is successful, make sure that the Local Traffic Manager active-standby units were configured correctly if you are migrating from a previous version.

Important: *During the upgrade, all users currently logged on to the system will have to log on again.*

1. Connect to a standby unit of a failover pair.
2. Upgrade the standby unit.
3. Press **Force offline** on the unit to trigger a failover to this newly upgraded unit.

   The newly upgraded unit will take over as the active unit.
4. Once the upgraded unit takes over as active, restart the upgraded unit.

   This extra step of additional restart is required to flush out any of the old sessions which may been introduced from the the previously active unit from an older version of the software.
5. Wait for the upgraded unit to come back up.
6. Once the upgraded unit becomes the active unit, bring the other unit back online by pressing **Release offline**.

   This unit is now the standby unit.
7. Upgrade the standby unit.

Configuring Single Sign-On with Access Policy Manager

What is Single Sign-On?

Access Policy Manager® provides a Single Sign-On (SSO) feature that leverages the credential caching and credential proxying technology.

Credential caching and proxying is a two-phase security approach that asks users to enter their credentials once to access their secured web applications. By leveraging this technology, users request access to the secured back-end web server. After that occurs, Access Policy Manager creates a user session and collects the user identity based on the access policy. When the access policy completes successfully, the user identity is saved (cached) in a session database. Access Policy Manager subsequently reuses the cached identity to seamlessly log the user into the secured web applications, thus providing the user with a single sign-on experience.

The Single Sign-On (SSO) feature provides the following benefits:

- Eliminates the need to administer and maintain multiple user logins
- Eliminates the need for users to enter their credentials multiple times.

Single Sign-On Methods

What are the supported SSO methods?

Access Policy Manager® supports the following SSO authentication methods.

SSO method	Description
HTTP Basic	Access Policy Manager uses the cached user identity and sends the request with the authorization header. This header contains the token `Basic` and the `base64-encoded` for the user name, colon, and the password.
HTTP Forms	Upon detection of the start URL match, Access Policy Manager uses the cached user identity to construct and send the HTTP form-based post request on behalf of the user.
HTTP Forms - Client Initiated	Upon detection of the request for logon page (URI, header, or cookie that is configured for matching the request), Access Policy Manager generates JavaScript code, inserts it into the logon page and returns the logon page to the client, where it is automatically submitted by inserted JavaScript. APM® processes the submission and uses the cached user identity to construct and send the HTTP form-based post request on behalf of the user.
HTTP NTLM Auth v1	NTLM employs a challenge-response mechanism for authentication, where the users can prove their identities without sending a password to the server.
HTTP NTLM Auth v2	NTLM employs a challenge-response mechanism for authentication, where the users can prove their identities without sending a password to the server. This version of NTLM is an updated version from NTLM v1.
Kerberos	This provides transparent authentication of users to Windows Web application servers (IIS) joined to Active Directory domain. It is used when IIS servers request Kerberos authentication; this SSO mechanism allows the user to get a Kerberos ticket and have Access Policy Manager present it transparently to the IIS application.
SAML	A SAML IdP service is a type of single sign-on (SSO) authentication service in Access Policy Manager that provides SSO authentication for external SAML service providers (SPs). You configure a SAML IdP service when you use a BIG-IP® system as a SAML identity provider (IdP).

About the Single Sign-On configuration object

Access Policy Manager® supports various SSO methods. Each method contains a number of attributes that you need to configure properly to support SSO.

Mis-configuring SSO objects for any of these authentication methods (HTTP Basic, NTLM v1 and v2, and Kerberos) could disable SSO for all authentication methods for a user's session when the user accesses a resource with the mis-configured object. The exceptions are Forms and Forms - Client Initiated, which are the only SSO methods that are not disabled when any other method fails due to a mis-configured SSO object.

Creating an HTTP Basic SSO configuration

With the HTTP Basic method of authentication, the SSO plug-in uses the cached user identity and sends the request with the authorization header. This header contains the Basic token and the base64-encoding of the user name, colon, and the password.

1. On the Main tab, click **Access** > **Single Sign-On** > **HTTP Basic**.
   The HTTP Basic screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5.
6. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
7. In the SSO Method Configuration area, specify the relevant settings.
8. Click **Finished**.

HTTP Basic SSO configuration settings

These settings are available when you create an HTTP Basic SSO configuration.

General Properties settings for HTTP Basic SSO configuration

Setting	Value	Additional Information
General Properties	**Basic** or **Advanced**. Defaults to **Basic**.	Additional settings are available when you select **Advanced**.
Name	Name of the SSO configuration.	The name must begin with a letter, or underscore, and contain only letters, numbers, underscores, dashes, and periods. Avoid using global reserved words in the name, such as all, delete, disable, enable, help, list, none, show, or None.
Headers	Header name-value pairs to send with the SSO method.	Available when you select **Advanced** from the **General Properties** list.

Credentials Source settings for HTTP Basic SSO configuration

Setting	Value	Additional Information
Username Source	Specifies the user name to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.username`
Password Source	Specifies the password to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.password`

SSO configuration settings for HTTP Basic SSO configuration

Setting	Value	Additional Information
Username Conversion	This check box is clear by default.	Select the check box to convert the PREWIN2k/UPN user name input format to the format you want to use for SSO. For example, convert `domain\username` or `username@domain` to `username`.

Creating an HTTP forms-based SSO configuration

With the HTTP forms method of authentication, upon detection of the start URL match, the SSO plug-in uses the cached user identity to construct and send the HTTP form-based POST request on behalf of the user.

1. On the Main tab, select **Access** > **Single Sign-On** > **Form Based**.
   The Form Based screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. From the **Use SSO Template** list, select the template you want to use.
   The screen refreshes to show additional settings applicable to the specific template.
5. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
6. If you selected **None** from the **Use SSO Template** list, fill in the relevant settings in the SSO Method Configuration area.

   Otherwise, these settings are taken from the template that you selected.
7.
8. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
9. Click **Finished**.

HTTP Form SSO configuration settings

These settings are available when you create an HTTP form-based SSO configuration.

General Properties settings for HTTP form-based SSO configuration

Setting	Value	Additional Information
General Properties	**Basic** or **Advanced**. Defaults to **Basic**.	Additional settings are available when you select **Advanced**.
Name	Name of the SSO configuration.	The name must begin with a letter, or underscore, and contain only letters, numbers, underscores, dashes, and periods. Avoid using global reserved words in the name, such as all, delete, disable, enable, help, list, none, show, or None.
Use SSO Template		If you select **None**, you must fill in the SSO Method Configuration area. Otherwise, the SSO Method Configuration area is not available; settings are configured with data supplied by the template you select.

Setting	Value	Additional Information
Headers	Header name-value pairs to send with the SSO method.	Available when you select **Advanced** from the **General Properties** list.

Credentials Source settings for HTTP form-based SSO configuration

Setting	Value	Additional Information
Username Source	Specifies the user name to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.username`
Password Source	Specifies the password to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.password`

SSO configuration settings for HTTP form-based SSO configuration

Setting	Value	Additional Information
Destination (IP or Hostname)	Defines the IP address or fully qualified domain name for the OWA server.	Displays only when you select an SSO template for OWA.
Start URI	Defines the start URI value. HTTP form-based authentication executes for SSO if the HTTP request URI matches the start URI value.	Multiple start URI values in multiple lines can be entered for this attribute. Supported session variable: `start_uri`
Pass Through	If you select the **Enable** check box, cookies presented in the form propagate to the client browser. Defaults to cleared.	
Form Method	Defines the SSO authentication method : **GET** or **POST**. Defaults to **POST**.	If you specify **GET**, the SSO authentication method is an HTTP GET request.
Form Action	Defines the form action URL used for HTTP authentication request for SSO.	For example, `/access/oblix/apps/webgate/bin/webgate.dll`. If left blank, the original request URL is used for SSO authentication. Supported session variable: `form_action`
Form Parameter For User Name	Defines the parameter name of the logon user name.	For example, the user ID is specified as the attribute value if the HTTP server expects the user name in the form of `userid=`. Supported session variable: `form_parameter`
Form Parameter for Password	Defines the name of the logon password.	For example, `Pass` is specified as the attribute value if the HTTP server expects the password in the form of `Pass`.

Setting	Value	Additional Information

Hidden Form Parameters/ Values	Defines the hidden form parameters required by the authentication server logon form at your location.	Specify a parameter name, a space, and the parameter value, if any. Each parameter must start on a new line. This example includes parameters for `platform` and `language`.
		`platform %{session.client.platform}`
		`language American English`
		Note: *When using a session variable as a value, precede it with a percent (%) sign and enclose it in curly braces ({}).*

Successful Logon Detection Match Type	Defines how Access Policy Manager® detects whether the user was successfully authenticated by the server. Defaults to **None**. You can select one option.	• **None** No check is made for authentication success. • **By Resulting Redirect URL** Authentication success is checked for by examining the redirect URL from the HTTP response. Multiple values can be specified for this option. • **By Presence Of Specific String in Cookie** Authentication success is checked for by searching for the string in the response. Supported session variable: `success_match_value`
Successful Logon Detection Match Value	Defines the value for the specific success detection type: the redirect URL or cookie name.	For **By Resulting Redirect URL** , you can specify multiple URLs and the system supports a single instance of the wildcard character (*) in a URL.

Creating an NTLMV1 SSO configuration

The NTLM authentication method employs a challenge-response mechanism for authentication, where the users can prove their identities without sending a password to a server.

1. On the Main tab, click **Access** > **Single Sign-On** > **NTLMV1**.
   The NTLMV` screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
6. In the SSO Method Configuration area, specify the relevant settings.
7. Click **Finished**.

NTLMV1 SSO configuration settings

These configuration settings are available when you configure an NTLMV1 SSO method.

General Properties settings for NTLMV1 SSO configuration

Setting	Value	Additional Information
General Properties	**Basic** or **Advanced**. Defaults to **Basic**.	Additional settings are available when you select **Advanced**.
Name	Name of the SSO configuration.	The name must begin with a letter, or underscore, and contain only letters, numbers, underscores, dashes, and periods. Avoid using global reserved words in the name, such as all, delete, disable, enable, help, list, none, show, or None.
Headers	Header name-value pairs to send with the SSO method.	Displayed when you select **Advanced** from the **General Properties** list.

Credentials Source settings for NTLMV1 SSO configuration

Setting	Value	Additional Information
Username Source	Specifies the user name to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.username`
Password Source	Specifies the password to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.password`
Domain Source	Specifies the domain to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.logon.last.domain`

SSO configuration settings for NTLMV1 SSO configuration

Setting	Value	Additional Information
Username Conversion	Check box is cleared by default.	Select the check box to convert the PREWIN2k/UPN user name input format to the format you want to use for SSO. For example, convert `domain\username` or `username@domain` to `username`.
NTLM Domain	Specifies the name of the domain where all users and groups are authenticated.	Specifies a domain name.

Creating an NTLMV2 SSO configuration

With this method of authentication, NTLM employs a challenge-response mechanism for authentication, where the users can prove their identities without sending a password to a server. This version of NTLM has been updated from version 1.

1. On the Main tab, click **Access** > **Single Sign-On** > **NTLMV2**.
   The NTLMV2 screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.

3. In the **Name** field, type a name for the SSO configuration.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
6. In the SSO Method Configuration area, specify the relevant settings.
7. Click **Finished**.

NTLMV2 SSO configuration settings

These configuration settings are available when you configure an NTLMV2 SSO method.

General Properties settings for NTLMV2 SSO configuration

Setting	Value	Additional Information
General Properties	Basic or **Advanced**. Defaults to **Basic**.	Additional settings are available when you select **Advanced**.
Name	Name of the SSO configuration.	The name must begin with a letter, or underscore, and contain only letters, numbers, underscores, dashes, and periods. Avoid using global reserved words in the name, such as all, delete, disable, enable, help, list, none, show, or None.
Headers	Header name-value pairs to send with the SSO method.	Displayed when you select **Advanced** from the **General Properties** list.

Credentials Source settings for NTLMV2 SSO configuration

Setting	Value	Additional Information
Username Source	Specifies the user name to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.username`
Password Source	Specifies the password to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.password`
Domain Source	Specifies the domain to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.logon.last.domain`

SSO configuration settings for NTLMV2 SSO configuration

Setting	Value	Additional Information
Username Conversion	Check box is cleared by default.	Select the check box to convert the PREWIN2k/UPN user name input format to the format you want to use for SSO. For example, convert `domain\username` or `username@domain` to `username`.

Setting	Value	Additional Information
NTLM Domain	Specifies the name of the domain where all users and groups are authenticated.	Specifies a domain name.

About NTLMv2 SSO failure for an invalid HTTP 401 response

If an HTTP 401 response from a server includes more than one `www-authenticate: NTLM` header, NTLMv2 SSO fails. This is expected behavior. An HTTP 401 response should contain only one `www-authenticate: NTLM` header.

Form-Based Client-Initiated Single Sign-On Method

About form-based client-initiated SSO authentication

With the HTTP form-based client-initiated method of authentication, when Access Policy Manager® detects the request for a logon page (URI, header, or cookie that is configured for matching the request), APM® generates JavaScript code, inserts it into the logon page, and returns the logon page to the client, where it is automatically submitted by the inserted JavaScript. APM processes the submission and uses the cached user identity to construct and send the HTTP form-based post request on behalf of the user.

Basic configuration of form-based client-initiated SSO

To create a form-based client-initiated SSO configuration object, you must configure at least one form and include at least one form parameter. A *form parameter* represents an input element on an HTML logon form, such as a form field for entering a user name or password, or, optionally, for entering a hidden form parameter.

Form-based client-initiated SSO configuration supports four sets of matching criteria that you can define.

Request Detection
(Required) Configures the SSO module to detect the HTTP request for the logon page by matching the HTTP URI, header, or cookie that you specify, and supports entry of multiple URIs. Requires data that is specific to the application. Request detection is successful when the request matches one of the configured items either partially or fully, depending on whether the request prefix option is enabled in Advanced Settings.

Form Identification
(Optional) Specifies how to detect the form within the HTTP body of the logon page. The default is form parameters, which enables identification of the logon form parameter fields based on the values entered for the form parameters in the general properties. Alternatively, you can specify that the form be identified using other data present in the form, such as the ID, name, or action attributes, or the form order.

Form Submit Detection
(Required) Specifies how to detect the submit request for the for a logon form. The default is an enabled auto detect option. Alternatively, you can select a scheme to use to use as an alternative to auto detect.

Logon Detection
(Optional) Configures the SSO module to detect whether logon was successful by checking for the presence of a cookie or a redirect URI. The default is **None** (logon detection is not performed).

The majority of web applications have a single logon page with one logon form. You need to define a single form for these applications. In less usual cases when an application has multiple logon pages with different logon forms, you need to create multiple forms, one for each logon page. If multiple logon pages use the same form, you need only one form with a list of URIs for all logon pages.

How does form-based client-initiated SSO authentication work by default?

This figure illustrates the default behavior of the form-based client-initiated SSO authentication method.

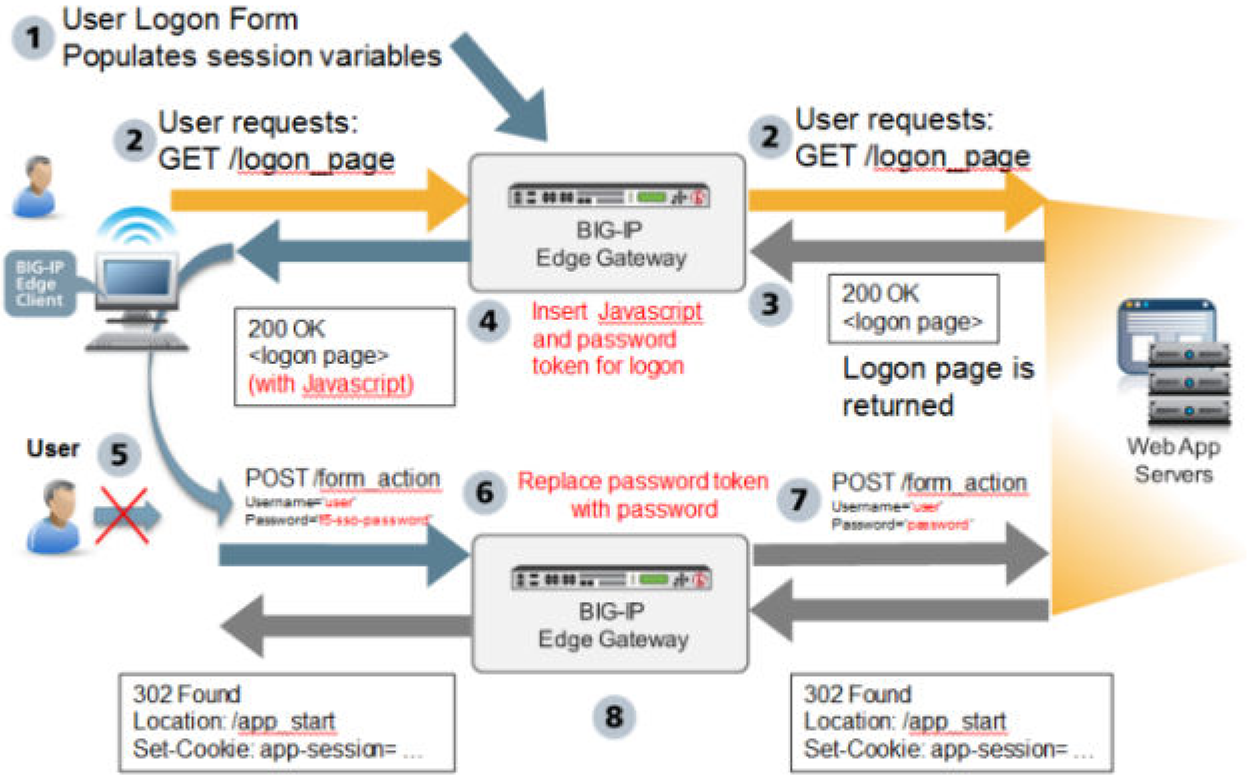


Figure 21: Form-based client-initiated SSO default behavior

1. The user logs on to Access Policy Manager® and APM® runs the access policy. This populates the session variables with the user credentials.
2. The user requests the application logon page. This GET request is passed to the application web server, verbatim.
3. The application web server replies with 200 OK and serves the logon page.
4. APM generates JavaScript and inserts it into the logon page before returning it to the user. The JavaScript assigns values to form parameters, as specified in the form configuration. The password parameter is assigned a password token rather than the actual user password.
5. The JavaScript runs on the client side. The logon page is not displayed to the user; user input is locked out. Without delay, the form is submitted using POST. The form parameters and their values, including user name and password token, are sent to APM.
6. APM then replaces the password token with the actual user password, as well as other form parameters specified in the form configuration with their configured values.
7. The POST, along with the real user credentials from step 1, is sent to the web server.
8. The application start page is served by the webserver, and sent to the client, verbatim. Optionally, APM performs detection of successful logon by examining HTTP response headers, looking for a cookie or redirect Location URI.

About advanced configuration options for form-based client-initiated SSO authentication

You can change some aspects of the form-based client-initiated SSO default behavior by configuring optional properties.

- You can change the automatically generated JavaScript code that is inserted into the logon page in one of three ways using the JavaScript Insertion options. You can replace it completely with custom code, or add extra code to it by specifying the application JavaScript functions to call prior to submitting a logon form.

- You can configure the SSO module to automatically detect the application HTTP request that submits user credentials using Form Submit Detection. If you disable automatic detection, the SSO module instead detects form submittal by using an HTTP header, cookie, or HTTP URIs that you specify.

Configuring form-based client-initiated SSO

You can use the form-based client-initiated SSO method to create form-based SSO configurations. For example, you can use this SSO method to support web applications that run JavaScript in the browser and need to maintain application state during the login process. You can also use it to support web applications that present multiple login screens.

1. On the Main tab, click **Access** > **Single Sign-On** > **Forms - Client Initiated**.
   The Forms - Client Initiated screen opens.
2. Click **Create**.
   A popup screen, Create New Forms-Client Initiated Configuration, opens.
3. In the **SSO Configuration Name** field, type a name.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5. If you want APM to gather and log information that you can use to configure form settings, select **Passthrough Configuration**.

   When you start a session using this SSO configuration, APM logs the information at the NOTICE level and prefixes it with PASSTHROUGH MODE LOG. You can view the logs in the sessions report. Reports are available in the **Access Overview Access Reports** area of the product.

   Important: *When* ***Passthrough Configuration*** *is selected, APM does not validate the form settings that you configure. This enables you to gather information, configure forms, and test them freely.*

   Important: *When you complete your testing, be sure to clear the* ***Passthrough Configuration*** *check box.*
6. If you selected **Passthrough Configuration** and you do not want to start configuring form settings now, click **OK**.

   The remainder of this procedure steps through configuring form settings.

   The new form-based client-initiated SSO configuration is available for testing.
7. Select **Form Settings** from the left pane.
8. Click **Create**.

   The **Create** button is not active until you complete the General Settings by typing a name for the SSO configuration.

   Note: *You must create at least one form to complete the SSO configuration (unless* ***Passthrough Configuration*** *is selected on the General Settings screen).*

   The Create New Form Definition popup screen opens.
9. Type a name in the **Form Name** field.
10. In the left pane, click **Request Detection**.
    The right pane displays required fields.
11. From the **Detect request for form by** list, select an option and type required data.

- **Cookie** Type a name in the **Cookie Name** field.
- **Header** Type a name in the **Header Name** field.
- **URI** Type a URI in the **Request URI** field.

The **OK** button becomes available.

12. In the Advanced Settings area, select an option for **Request Method**.

    Specifies whether the request method is **GET** or **POST**. Defaults to **GET**.
13. Select **Form Identification** from the left pane.
    Create New Form Definition displays in the right pane.
14. From the **Identify Form by** list, select how to find the HTML logon form in the HTML body of the logon page.
15. Select **Form Parameters** from the left pane.
    Form Parameters displays in the right pane.
16. For each form parameter that you want to create, repeat these steps:
    a) Click **Create**.
       The Create New Form Parameter popup screen opens.
    b) In the **Form Parameter Name** field, type or select a name.
    c) In the **Form Parameter Value** field, type or select a value.
    d) For the **Secure** option, select **Yes** if applicable.
    e) Click **OK**.

    The screen closes, showing the Create New Form Definition popup screen, which displays the new form parameter.
17. Select **Form Submit Detection** from the left pane.
    The Create New Form Definition popup screen opens.
18. For **Disable Auto detect submit**, retain the default value, **No**.
19. Select **Logon Detection** from the left pane.
20. From **Detect Login by**, select an option for detecting a successful login and type any required data:

    - **None**.
    - **Presence of Cookie** Type a name in the **Cookie Name** field.
    - **Redirect URI** Type a URI in the **Redirect URI** field.
21. Click **OK**.
    The screen closes, displaying the Forms - Client Initiated screen for SSO Configurations.

The new form-based client-initiated SSO configuration is available for use.

Forms-based client-initiated SSO configuration settings

These settings are available when you create a form-based client-initiated SSO configuration.

General settings

Setting	Description
SSO Configuration Name	Specifies the name of the configuration. It must be unique.
Passthrough Configuration	This option helps administrators configure SSO **Form Settings**. **Form Settings** are not mandatory when this option is enabled. When starting a session with SSO passsthrough enabled, the relevant form settings information is logged in the session report. Disable **Passthrough Configuration** after configuring **Form Settings** correctly with the help of the session passthrough logs.
SSO Description	Specifies a description. This is an optional setting.

Setting	Description
Log Settings	Specifies at what level of detail the system logs. Valid values are listed. Defaults to **Notice**.

Form settings

Table 1: General Properties

Setting	Description
Form Name	Specifies the name of the form. It can be any name and need not match the actual name of the HTML form.
Form Description	Specifies an optional description of the form.

Table 2: Request Detection

Setting	Description
Detect request for form by	Specifies which element of the HTTP request headers is used to identify the application request for logon page: Cookie, Header, or URI. Defaults to URI.
Cookie	Specifies that the system identifies the form by the presence (default) or absence (configurable with Advanced Properties) of this cookie.
Header	Specifies that the system identifies the form by the presence (default) or absence (configurable with Advanced Properties) of a header.
URI	Specifies that the system identifies the form by a successful match (default) or failed match (configurable with Advanced Properties) against one or multiple URIs.

Table 3: Advanced Settings - Request Detection

Setting	Description
Request Method	Specifies whether the request method is **GET** or **POST**. Defaults to **GET**.
Request Negative	When selected, specifies that the system detects the form that fails to match the criteria specified for Form Detection. The system then detects the form by the absence of the specific cookie or header, or by its failure to match the URIs. The default is cleared.
Request Prefix	When selected, specifies that the system matches on a partial string. If this option is not selected, the match must be verbatim. The default is selected.

Table 4: Form Identification

Setting	Description
Identify Form by	Specifies how the HTML logon form is found in the HTML body of the logon page. If there is more than one form on the logon page matching the criteria, the first match is used. Options are:

Setting	Description
	• **ID Attribute-**Specifies that a form ID is used to find the form. • **Name Attribute**-Specifies that • **Action Attribute**-Specifies that • **Form Order**-Specifies that • **Form Parameters** (default)--Specifies that the form parameters, which have already been defined, are used to find the form. There is nothing more to configure.
Form ID	Specifies the form ID that is used to identify the form.
Form Name	Specifies the specific form name.
Form Action	Specifies the value of the action attribute.
Form Order	Specifies the relative order of the form on the logon page (starting from 1).
Form Parameters	Specifies the name and value of the form parameter and whether the parameter is encrypted.

Table 5: Form Parameters

Setting	Description
Form Parameter Name	Specifies the name of a form parameter.
Form Parameter Value	Specifies the value of the form parameter. This is usually the name of a session variable. The value could also be a literal string or a combination of strings and session variable names. ***Note:*** *If the session variable is not found when the SSO request is processed, the value of the corresponding POST parameter will be empty.*
Secure	Specifies whether the parameter is secure. Defaults to **No**.

Table 6: Form Submit Detection

Setting	Description
Disable Auto detect submit	Defaults to **No.**
Scheme	Available when Disable Auto detect submit is set to Yes. Specifies how to detect submit. Options are: • **URI** • **Cookie** • **Header**

Table 7: Advanced Settings - Form Submit Detection

Setting	Description
Submit Request Negative	When selected, specifies that the system detects the form that fails to match the criteria specified for Form Detection. The system then detects the form by the absence of the specific cookie or header or by its failure to match the URIs. The default is cleared.

Setting	Description
Submit Request Prefix	When selected, specifies that the system matches on a partial string. If this option is not selected, the match must be verbatim. The default is selected.

Table 8: Logon Detection

Setting	Description
Detect Login by	Specifies whether and how to detect a successful logon. Options are: • **Presence of Cookie** • **Redirect URI** • **None** (default)
Cookie Name	Specifies the cookie name that identifies successful logon.
Redirect URI	Specifies the redirect URI that identifies successful logon.

Table 9: JavaScript Injection

Setting	Description
Injection Method	Specifies whether to use the default JavaScript that APM™ creates. Defaults to Auto. • Auto • Extra • Custom
Extra Javascript	Specifies more JavaScript to run at the end of the automatically generated JavaScript. ***Note:*** *Review the logon page source to determine whether any JavaScript functions are called on submit.*
Custom Javascript	Specifies the custom JavaScript to run in place of the automatically generated JavaScript. When you select the **Custom** injection method, a JavaScript template is provided in the **Custom Javascript** text area. You must modify this in order to add the appropriate form parameters.

Header Settings

Setting	Description
Header Name	Name
Header Value	Value

Form-based client-initiated SSO configuration examples

Using the examples provided for various applications, you can quickly create form-based client-initiated SSO configurations.

DWA form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Domino Web Access (DWA).

Setting	Sample value
SSO Configuration Name	`ssov2-dwa`
Form Name	`testform`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Username` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	/
Identify Form by	`Name Attribute`
Form Name	`STLogonForm`
Detect Logon by	`Presence of Cookie`
Cookie Name	`DomAuthSessId`
Request Prefix	Not selected

Bugzilla form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Bugzilla.

Setting	Sample value
SSO Configuration Name	`ssov2-bugzilla`
Form Name	`tform`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Bugzilla_login` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Bugzilla_password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	**URI**
Request URI	`/`
Identify Form by	**ID Attribute**
Form ID	`mini_login_top`
Detect Logon by	**Presence of Cookie**

Setting	Sample value
Cookie Name	`Bugzilla_logincookie`
Request Prefix	Not selected

Ceridian form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Ceridian.

Settings to configure form-based client-initiated SSO for Ceridian

Setting	Sample value
SSO Configuration Name	`ssov2_ceridian`
SSO Description	`sourcetimepro1.ceridian.com`
Form Name	`auth_form`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `ClientIDInput` • `%{session.logon.last.clientid}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `SerialNumberInput` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `PasswordInput` • `%{session.sso.custom.last.password}` • **No** (Default)
Detect Form by	**URI**

Request URI	`/`
	`/sta.asp`
	`/ctagw/`
	`/ctagw/sta.asp`

Identify Form by	**Form Parameters**
Detect Logon by	**Redirect URI**
Redirect URI	`https://sourcetimepro1.ceridian.com/CTA660/cta.asp?RequestID=*`
Request Prefix	Not selected
Injection Method	**Custom**
Custom Javascript	See sample code that follows.
Disable Auto detect submit	**Yes**
Scheme	**URI**
URI	`/sta.asp` `/ctagw/sta.asp`

Custom JavaScript

```
<script>
function checkInternetExplorerVersion()
// Returns 'true' if the version of Internet Explorer > 8
{
  var r = -1; // Return value assumes agreement.
  if (navigator.appName == 'Microsoft Internet Explorer')
  {
    var ua = navigator.userAgent;
    var re  = new RegExp("MSIE ([0-8]{1,}[\.0-9]{0,})");
    if (re.exec(ua) != null)
      r = parseFloat( RegExp.$1 );
  }
  return ( r==-1 ) ? true : false;
}
if (checkInternetExplorerVersion()) {
  document.body.style.visibility='hidden';
  document.body.style.display='none';
}
document.body.onkeydown=function(e){return false;};
function __f5submit() {
var __f5form = document.forms[0];
__f5form.SerialNumberInput.value='%{session.sso.token.last.username}';
__f5form.PasswordInput.value='%{session.sso.custom.last.password}';
__f5form.ClientIDInput.value='%{session.logon.last.clientid}';
f_submit();
}
if (window.addEventListener) {
  window.addEventListener('load',__f5submit,false);
} else if (window.attachEvent) {
  window.attachEvent('onload',__f5submit);
} else {
  window.onload=__f5submit;
}
</script>
```

Logon Page customization in access policy

Logon Page Agent (field 3):

- **Type:** `text`
- **Post Variable Name:** `clientid`
- **Session Variable Name:** `clientid`

Logon Page Input Field #3: `Company ID`

Variable Assign definition in access policy

```
session.sso.custom.last.password = expr { [mcget -secure
{session.sso.token.last.password}] }
```

Citrix form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for some of the Citrix server product versions that F5® supports. For Citrix compatibility information, see the BIG-IP® APM® Client Compatibility Matrix on the AskF5™ web site at `http://support.f5.com/`.

Setting	Sample value
SSO Configuration Name	`sso_fbv2`
Form Name	`testform`

Setting	Sample value
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `domain` • `%{session.logon.last.domain}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `user` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	**URI**

Request URI	`/Citrix/AccessPlatform/auth/login.aspx`
	`/Citrix/XenApp/auth/login.aspx`
	`/Citrix/StoreWeb/Authentication/LoginAttempt`
	`/Citrix/StoreWeb/ExplicitAuth/Login`

Identify Form by	**Action Attribute**
Form Action	`login.aspx`
Detect Logon by	**Redirect URI**

Redirect URI	`*/Citrix/XenApp/site/default.aspx`
	`*/Citrix/AccessPlatform/site/default.aspx`
	`*/Citrix/StoreWeb/site/default.aspx`

Citrix Product Upgrades

Warning: *When you upgrade from one Citrix product version to another, it is not unusual for the product URIs to change. When that happens, form-based client-initiated SSO will stop working until you update the SSO configuration with the new URIs for the logon form and for redirect.*

Devcentral form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Devcentral.

Settings to configure form-based client-initiated SSO for Devcentral

Table 10: Devcentral Configuration Example

Setting	Sample value
SSO Configuration Name	`ssov2_devcentral`
SSO Description	`devcentral.f5.com`
Form Name	`auth_form`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `dnn$ctr1093548$Login$Login_DNN$cmdLogin` • `Login` • **No** (Default)

Setting	Sample value
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `dnn$ctr1093548$Login$Login_DNN$txtUsername` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `dnn$ctr1093548$Login$Login_DNN$txtPassword` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	`/Community/Login/tabid/1082224/Default.aspx` `/tabid/1082224/Default.aspx`
Identify Form by	`Form Parameters`
Detect Logon by	`Cookie`
Cookie Name	`authentication`
Injection Method	`Extra`
Extra Javascript	See sample code that follows.

Extra Javascript

```
WebForm_DoPostBackWithOptions(new WebForm_PostBackOptions("dnn$ctr1093548$Login$Login_DNN
$cmdLogin", "", true, "", "", false, false));
__f5form.enctype = 'application/x-www-form-urlencoded';
__f5form.encoding = 'application/x-www-form-urlencoded';
```

Google form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Google.

Setting	Sample value
SSO Configuration Name	`ssov2_google`
Description	`accounts.google.com`
Form Name	`form_auth`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Email` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `Passwd` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	**URI**
Request URI	`/ServiceLogin`
Identify Form by	**Form Parameters**
Detect Logon by	**Presence of Cookie**

Setting	Sample value
Cookie Name	`SID`

Note: *For Internet Explorer 7 (and 8), disable the advanced setting* ***Display a notification about every script error****.*

Oracle Application Server form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Oracle 10g Release 2 (10.1.2).

Setting	Sample value
SSO Configuration Name	`ssov2_oracle`
Form Name	`tform`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `ssousername` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	`/sso/pages/login.jsp?site2pstoretoken=v1.2`
Identify Form by	`Form Parameters`
Detect Logon by	`Cookie`
Cookie Name	`SSO_ID`

OWA 2010 and 2007 form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Outlook Web App (OWA) 2010 and OWA 2007.

Table 11: OWA 2010 and OWA 2007 Configuration Example

Setting	Sample value
SSO Configuration Name	`ssov2-owa`
Form Name	`tform`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `username` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`

Setting	Sample value
Request URI	/owa/auth/logon.aspx?replaceCurrent=1&url= /owa/auth/logon.aspx?url=
Identify Form by	Form Parameters
Detect Logon by	Presence of Cookie
Cookie Name	sessionid
Injection Method	Extra
Extra Javascript	clkLgn()

OWA 2003 form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Outlook Web App (OWA) 2003.

Setting	Sample value
SSO Configuration Name	ssov2-owa2003
Form Name	tform2003
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• username • %{session.sso.token.last.username} • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• password • %{session.sso.token.last.password} • **Yes**
Detect Form by	URI
Request URI	/exchweb/bin/auth/owalogon.asp?url=https:// ata.bldg12.grpy.company.com/exchange/&reason=0
Identify Form by	Form Parameters
Detect Logon by	Presence of Cookie
Cookie Name	sessionid

Perforce form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Perforce.

Setting	Sample value
SSO Configuration Name	perforce-sso
Form Name	p4
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• u • %{session.sso.token.last.username} • **No** (Default)
• **Form Parameter Name**	• p

Setting	Sample value
• **Form Parameter Value** • **Secure**	• `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	`/p4web`
Identify Form by	`Form Parameters`
Detect Logon by	`Presence of Cookie`
Cookie Name	`P4W8080`
Request Prefix	Not selected

Reviewboard form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Reviewboard.

Setting	Sample value
SSO Configuration Name	`reviewboard-sso`
Form Name	`rb_logon`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `username` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `password` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	`/account/login`
Identify Form by	`Form Parameters`
Detect Logon by	`Redirect URI`
Redirect URI	`*/dashboard`
Request Prefix	Not selected

SAP form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for SAP.

Setting	Sample value
SSO Configuration Name	`ssov2_sap`
Form Name	`tform`
• **Form Parameter Name** • **Form Parameter Value**	• `j_user` • `%{session.sso.token.last.username}`

Setting	Sample value
• **Secure**	• **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `j_password` • `%{session.sso.token.last.password}` • **Yes**
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `uidPasswordLogon` • `Log On` • **No** (Default)
Detect Form by	`URI`
Request URI	`/irj/portal`
Identify Form by	`Form Parameters`
Detect Logon by	`Presence of Cookie`
Cookie Name	`MYSAPSSOV2`
Request Prefix	Not selected

Salesforce form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Salesforce.

Setting	Sample value
SSO Configuration Name	`ssov2_salesforce`
Form Name	`auth_form`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `username` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `pw` • `%{session.sso.token.last.password}` • **Yes**
Detect Form by	`URI`
Request URI	`/`
Identify Form by	`Form Parameters`
Detect Logon by	`Cookie`
Cookie Name	`inst`
Injection Method	`Custom`
Custom Javascript	See sample code that follows.

Custom Javascript

```
<script>
      function checkInternetExplorerVersion()
```

```
// Returns 'true' if the version of Internet Explorer > 8
{
var r = -1; // Return value assumes agreement.
if (navigator.appName == 'Microsoft Internet Explorer')
{
var ua = navigator.userAgent;
var re  = new RegExp("MSIE ([0-8]{1,}[\.0-9]{0,})");
if (re.exec(ua) != null)
r = parseFloat( RegExp.$1 );
}
return ( r==-1 ) ? true : false;
}
if (checkInternetExplorerVersion()) {
document.body.style.visibility='hidden';
document.body.style.display='none';
}
document.body.onkeydown=function(e){return false;};
function __f5submit() {
var __f5form = document.forms[0];
__f5form.username.value='%{session.sso.token.last.username}';
__f5form.password.value='f5-sso-token';
;
var __f5action = __f5form.action;
var __f5qsep = (__f5action.indexOf('?') == -1) ? '?' : '&';
__f5form.action = __f5action + __f5qsep + 'f5-sso-form=auth_form';
__f5form.Login.click();
}
if (window.addEventListener) {
window.addEventListener('load',__f5submit,false);
} else if (window.attachEvent) {
window.attachEvent('onload',__f5submit);
} else {
window.onload=__f5submit;
}
</script>
```

Sharepoint 2010 form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Sharepoint.

Setting	Sample value
SSO Configuration Name	`ssov2_shp2010`
Form Name	`form_auth`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `ctl00$PlaceHolderMain$signInControl$UserName` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `ctl00$PlaceHolderMain$signInControl$password` • `%{session.sso.token.last.password}` • **Yes**
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `ctl00$PlaceHolderMain$signInControl$login` • `Sign In` • **Yes**
Detect Form by	`URI`
Request URI	`/_forms/default.aspx?ReturnUrl=`
Identify Form by	`Form Parameters`

Setting	Sample value
Detect Logon by	`Cookie`
Cookie Name	`FedAuth`

Weblogin form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Weblogin.

Setting	Sample value
SSO Configuration Name	`ssov2-weblogin`
Form Name	`tform`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `user` • `%{session.sso.token.last.username}` • **No** (Default)
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `pass` • `%{session.sso.token.last.password}` • **Yes**
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `submit_form` • `Submit` • **No** (Default)
Detect Form by	`URI`
Request URI	`/sso/login.php?redir=`
Identify Form by	`Name Attribute`
Form Name	`theForm`
Detect Logon by	`Cookie`
Cookie Name	`issosession`

Yahoo form-based client-initiated SSO example

This example lists settings and values for creating a form-based client-initiated SSO configuration for Yahoo.

Setting	Sample value
SSO Configuration Name	`sso_yahoo`
SSO Description	`login.yahoo.com`
Form Name	`form_login`
• **Form Parameter Name** • **Form Parameter Value** • **Secure**	• `login` • `%{session.sso.token.last.username}` • **No** (Default)
Detect Form by	`URI`
Request URI	`/`

Setting	Sample value
Identify Form by	`ID Attribute`
Form ID	`login_form`
Detect Logon by	`Cookie`
Cookie Name	`PH`
Injection Method	`Custom`
Custom Javascript	See example custom Javascript that follows.
Disable Auto detect submit	Selected
Javascript	`/config/login`

Custom Javascript

```
<script>
 //Logon page will not be hidden in IE7/8.
 //This is workaround for the problem with JS method .focus()
 //"Can't move focus to the control because it is invisible, not enabled, or of a type that
does not accept the focus."
function checkInternetExplorerVersion()
// Returns 'true' if the version of Internet Explorer > 8
{
  var r = -1; // Return value assumes agreement.
  if (navigator.appName == 'Microsoft Internet Explorer')
  {
    var ua = navigator.userAgent;
    var re  = new RegExp("MSIE ([0-8]{1,}[\.0-9]{0,})");
    if (re.exec(ua) != null)
      r = parseFloat( RegExp.$1 );
  }
  return ( r==-1 ) ? true : false;
}
if (checkInternetExplorerVersion()) {
  document.body.style.visibility='hidden';
  var inter = setInterval(function ()
  {
    var err = document.getElementsByClassName('yregertxt')[0];
    var wcl = document.getElementById('captcha_c');
    if (err) {
      document.body.style.visibility = 'visible';
      clearInterval(inter);
    }
    if (wcl) {
      if ( wcl.style.visibility == 'hidden') {
        document.body.style.visibility = 'visible';
        clearInterval(inter);
      }
    }
  }, 1000);
};
function __f5submit() {
var adv = document.getElementById('adFrame');
if (adv) adv.style.visibility='hidden';
var __f5form = document.forms[0];
if (__f5form.login)
  __f5form.login.value='%{session.sso.token.last.username}';
__f5form.passwd.value='%{session.sso.custom.last.password}';
__f5form[".save"].click();
}
if (window.addEventListener) {
  window.addEventListener('load',__f5submit,false);
} else if (window.attachEvent) {
  window.attachEvent('onload',__f5submit);
} else {
  window.onload=__f5submit;
}
</script>
```

Variable Assign definition used in access policy

```
session.sso.custom.last.password = expr { [mcget -secure
{session.sso.token.last.password}] }
```

Kerberos Single Sign-On Method

About Kerberos SSO

Access Policy Manager® provides seamless authentication to application servers (web servers) using Kerberos SSO. It is the only SSO method that can be used when authentication methods used by the access policy do not provide the user's password in clear text. Examples of such methods include client certificate authentication, NTLM authentication, or any other challenge/response authentication method where the password is not transmitted in clear text. To use Kerberos SSO, you must have Kerberos implemented in your environment, such as using Active Directory domain with IIS servers configured for Integrated Windows authentication.

How does Kerberos SSO work in Access Policy Manager?

You can leverage Kerberos SSO in the following ways:

- Using a virtual server with an access policy associated with it.
- Handling the SSO event through the use of Portal Access Resource. In this scenario, the Portal Access resource is assigned to the Access Policy and the virtual server attaches a rewrite profile.

Here is a typical scenario showing what occurs when Kerberos SSO is used if client certificate authentication is present:

1. When a user connects to the virtual server, Access Policy Manager® validates the credentials and extracts the UPN from the certificate through the access policy.
2. When the client accesses an application that requires a Kerberos ticket, the UPN and the configured Kerberos SSO object are used to retrieve the ticket from Active Directory. The ticket is then cached for the particular client and presented to the application for access.

Important: *Under other circumstances, the access policy may not ask for credentials within the certificate, because, for example, a logon page may be present. In such a case, the user name supplied by the client is used at the UPN. Other factors, such as the use of other types of authentication methods, must be present in order to ensure that the credentials are valid in order to retrieve the Kerberos ticket.*

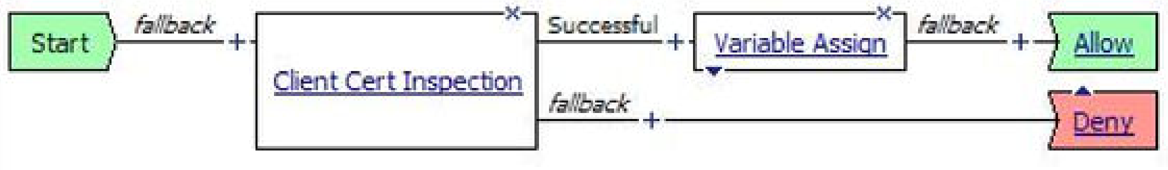


Figure 22: Example access policy for Kerberos SSO

Task summary for configuring Kerberos SSO

Access Policy Manager® lets you configure for Kerberos SSO.

To set up this configuration, follow the procedures in the task list.

Task List

Setting up a delegation account to support Kerberos SSO
Creating a Kerberos SSO configuration in APM
Editing an access policy to support Kerberos SSO
Binding a Kerberos SSO object to an access profile
Verifying log settings for the access profile
Attaching an access profile to a virtual server for Kerberos SSO

Setting up a delegation account to support Kerberos SSO

Before you can configure Kerberos SSO in Access Policy Manager®, you must create a delegation account in Active Directory.

Note: *For every server realm, you must create a delegation account in that realm.*

1. Open the Active Directory Users and Computers administrative tool and create a new user account.

   The user account should be dedicated for delegation, and the **Password never expires** setting enabled.
2. Set the service principal name (SPN) on the Windows server for the user account.

   For the support tools that you can use, and for the commands, such as `setspn` and `ktpass`, refer to Microsoft documentation.

   Note: *If you use the* `ktpass` *command, it sets the SPN on the Windows server and creates a keytab file. APM Kerberos SSO does not need or use a keytab file.*
3. Verify the result of setting the SPN.

   This example is purely for illustration. Refer to Microsoft documentation for up-to-date commands and correct usage.

   ```
   C:\Users\Administrator> setspn -L apm4
   ```

   ```
   Registered ServicePrincipalNames for
   CN=apm4,OU=users,DC=yosemite,DC=lab,DC=dnet,DC=com: HTTP/
   apm4.yosemite.lab.dnet.com
   ```
   where *`apm4`* is the name of the user account that you created.
4. Return to the Active Directory Users and Computers screen to open your account again.

   A Delegation tab should appear.
5. Click the Delegation tab.
6. Select **Trust this user for delegation to specified services only**.
7. Select **Use any authentication protocol**, and add all your services to the list under **Services to which this account can present delegated credentials**.

   Every service should have Service Type HTTP (or http) and host name of the pool member or web application resource host that you will use in your configuration.
8. Click **OK**.
   This creates the new delegation account.

Creating a Kerberos SSO configuration in APM

Before you start, you must have configured a delegation account in Active Directory.

To support Kerberos single sign-on authentication from Access Policy Manager® (APM®), you must create a Kerberos SSO configuration.

Note: *To complete this task, you need to know the service principal name (SPN) for the delegation account.*

1. On the Main tab, click **Access** > **Single Sign-On** > **Kerberos**.
   The Kerberos screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
5. In the Credentials Source area, specify the credentials that you want cached for Single Sign-On.
6. In the **Kerberos Realm** field, type the name of the realm in uppercase.

   For example, `MY.HOST.LAB.MYNET.COM`
7. In the **Account Name** field, type the name of the Active Directory account configured for delegation.

   Type the account name in SPN format.

   In this example `HTTP/apm4.my.host.lab.mynet.com@MY.HOST.LAB.MYNET.COM`, apm4 is the delegation account, apm4.my.host.lab.mynet.com is its fully qualified domain name, and MY.HOST.LAB.MYNET.COM is the realm.
8. In the **Account Password** and **Confirm Account Password** fields, type the delegation account password.
9. Click **Finished**.

Editing an access policy to support Kerberos SSO

After you create an access profile to support Kerberos SSO, you must edit the policy and add the appropriate agents.

Note: *These steps walk you through creating an example access policy that takes information from the client certificate to populate the session variables that Kerberos SSO uses.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. From the list, select an access profile to which you want to add Kerberos SSO support.
   The properties screen for that access profile opens.
3. On the menu bar, click **Access Policy**.
   Access policy settings display.
4. In the General Properties area, click the **Edit Access Policy for Profile** ***`profile_name`*** link.
   The visual policy editor opens the access policy in a separate screen.
5. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
6. On a policy branch, click the **(+)** icon to add an item to the policy.
   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
7. From Authentication, select **Client Cert Inspection**, and click **Add item**.

A properties screen opens.

8. Click **Save**.
   The properties screen closes and the policy displays.
9. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
10. On a policy branch, click the **(+)** icon to add an item to the policy.
    A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
11. From **General Purpose**, select **Variable Assign** and click **Add item**.
    A properties screen opens.
12. Add an entry to the Variable Assign properties to extract the UPN from the client certificate Subject Alternative Name field:

    a) Click **Add new entry**.
    An **empty** entry appears in the Assignment table.

    b) Click the **change** link next to the empty entry.
    A dialog box opens, where you can enter a variable and an expression.

    c) On the left side, retain the selection of **Custom Expression** and, in the field, type a variable name, such as: `session.logon.last.upn`.

    d) On the right side, in the field type an expression to extract the UPN from the client certificate:
    For example, type the following expression.

```
set e_fields [split  [mcget {session.ssl.cert.x509extension}] "\n"];
foreach qq $e_fields {
  if {[string first "othername:UPN" $qq] >= 0} {
    return [string range $qq [expr { [string first "<" $qq] + 1 } ] [expr { [string first
">" $qq] - 1 } ] ];
  }
}
return "";
```

    e) Click **Finished**.

    The popup screen closes.

13. Add another entry to populate the `session.logon.last.username` variable by extracting it from the UPN:

    a) Click **Add new entry**.
    An **empty** entry appears in the Assignment table.

    b) Click the **change** link next to the empty entry.
    A dialog box opens, where you can enter a variable and an expression.

    c) On the left side, retain the selection of **Custom Expression** and, in the field, type this variable name: `session.logon.last.username`.

    d) On the right side, in the field type an expression to extract the user name from the UPN:
    For example, type the following expression.

```
set upn [mcget {session.logon.last.upn}];
if {[string first "@" $upn] >= 0} {
  return [string range $upn 0 [expr { [string first "@" $upn] - 1 } ] ];
} else {
  return $upn;
}
```

    e) Click **Finished**.

The popup screen closes.

14. Add another entry to populate the `session.logon.last.domain` variable with the realm by extracting it from the UPN:
    a) Click **Add new entry**.
    An **empty** entry appears in the Assignment table.
    b) Click the **change** link next to the empty entry.
    A dialog box opens, where you can enter a variable and an expression.
    c) On the left side, retain the selection of **Custom Expression** and, in the field, type this variable name: `session.logon.last.domain`.
    d) On the right side, in the field type an expression to extract the realm from the UPN:
    For example, type the following expression.

```
set upn [mcget {session.logon.last.upn}];
if {[string first "@" $upn] >= 0} {
  return [string range $upn [expr { [string first "@" $upn] + 1 } ] end ];
} else {
  return "";
}
```

    e) Click **Finished**.

    The popup screen closes.
15. Click **Save**.
    The properties screen closes and the policy displays.

You have created an access policy to support Kerberos SSO.

The next step is to bind the SSO object to the access profile.

Binding a Kerberos SSO object to an access profile

Before beginning this task, configure an SSO object with Kerberos authentication or ensure that such an SSO object exists.

To bind a Kerberos SSO object to an access profile, add an SSO configuration (Kerberos SSO object) to it.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. From the list, select an access profile to which you want to add Kerberos SSO support.
   The properties screen for that access profile opens.
3. Click the **SSO Auth/Domains** tab.
4. From the **SSO Configuration** list, select an SSO configuration with Kerberos authentication that you previously identified or configured.
5. Click **Update**.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.

2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Attaching an access profile to a virtual server for Kerberos SSO

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address in CIDR format.

   The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is `10.0.0.1` or `10.0.0.0/24`, and an IPv6 address/prefix is `ffe1::0020/64` or `2001:ed8:77b5:2:10:10:100:42/64`. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a `/32` prefix.

   Note: *The IP address you type must be available and not in the loopback network.*
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured for Kerberos SSO.
7. Click **Finished**.

Kerberos SSO configuration settings

These settings are available when you configure a Kerberos SSO method.

General Properties settings for Kerberos SSO configuration

Setting	Value	Additional Information
General Properties	Basic or **Advanced**. Defaults to Basic.	Additional settings are available when you select **Advanced**.
Name	Name of the SSO configuration.	The name must begin with a letter, or underscore, and contain only letters, numbers, underscores, dashes, and periods. Avoid using global reserved words in the name, such as all, delete, disable, enable, help, list, none, show, or None.
Headers	Header name-value pairs to send with the SSO method.	Displayed when you select **Advanced** in the **General Properties** list.

Credentials Source settings for Kerberos SSO configuration

Setting	Value	Additional Information
Username Source	Specifies the user name to cache for single sign-on. Defaults to a session variable.	Supported session variable: `session.sso.token.last.username`
User Realm Source	Displays the session variable, if configured, that specifies the realm for the user. If the variable is set, it must contain the Kerberos realm for the user.	If this field is left empty or the variable does not exist or has no value, the user is assumed to be in the same Kerberos realm as the server. Supported session variable: `session.logon.last.domain`

SSO configuration settings for Kerberos SSO configuration

Setting	Value	Additional Information
Kerberos Realm	Specifies the realm of application servers, such as pool members or portal access resource hosts.	If servers are located in multiple realms, you must create a separate SSO configuration for each realm. Realm must be specified in uppercase letters or can be specified using the *`session.logon.last.domain`* session variable. ***Note:*** *The **KeepAlive** setting on your backend webserver must be enabled for Kerberos authentication to work properly.*
KDC	Specifies the IP Address or the host name of the Kerberos Key Distribution Center (KDC) (normally an Active Directory domain controller) for the server realm.	***Note:*** *KDC must be empty when the user realm is different from the server realm and in the case of multi-domain realms.* If KDC is empty, the KDC must be discoverable through DNS. For example, the BIG-IP® system must be able to fetch SRV records for the server realm's domain, where the domain name is the same as the realm name. If the domain name is different from the realm name, it must be specified in the `/etc/krb5.conf` file. Kerberos SSO processing is fastest when KDC is specified by its IP address, slower when specified by host name, and, due to additional DNS queries, even slower when empty.
Account Name	Specify the name of the Active Directory account configured for delegation.	This account must be configured in the Kerberos realm (AD Domain) of the server. ***Note:*** *If servers are from multiple realms, each realm (AD Domain) must have its own delegation account.*
Account Password	Specifies the password for the delegation account specified in the **Account Name** field.	

Setting	Value	Additional Information
Confirm Account Password	Verifies the password specified in the **Account Password** field.	
SPN Pattern	An optional field for modifying how the Service Principal Name (SPN) for the servers is constructed.	Leave this field empty unless you need a non-standard SPN format. The default value for this field is HTTP/%s@`REALM`, where %s is replaced by the server host name discovered through reverse DNS lookup using the server IP address. When entering a string, replace REALM with an actual realm name (as specified in Kerberos Realm setting).
Ticket Lifetime	Represents the maximum ticket lifetime in minutes. Defaults to 600. Minimum is 10.	Should not be set higher than the value configured for the Active Directory delegation account (which defaults to 600). ***Note:*** *The actual lifetime can be less than the configured value by up to 1 hour because the user's ticket lifetime is the same as the Kerberos Ticket Granting Ticket (TGT) ticket lifetime.* The TGT for the delegation account specified in this configuration is obtained. A new TGT is fetched every time the latest TGT is older than one hour, but only when an SSO request is processed.
Send Authorization	Specifies when to submit the Kerberos ticket to application servers: **Always** or **On 401 Status Code**. Defaults to **Always**.	The Kerberos ticket is submitted in the HTTP Authorization header. The header value starts with the word Negotiate, followed by one space and a base64 encoded GSSIAPI token that contains the Kerberos ticket. If the request contains an Authorization header from the client browser, it is deleted. The options are defined here. • **Always** The Authorization header with a Kerberos ticket is inserted into every HTTP request whether or not it requires authentication; in other words, it is inserted preemptively. The Kerberos ticket GSSAPI representation uses KRB5 Kerberos 5 mechanism displays (OID 1.2.840.113554.1.2.2). Selecting **Always** results in the additional overhead of generating a Kerberos token for every request. Kerberos tickets are fetched for first request only for the user and then cached for up to the configured ticket lifetime, so that subsequent requests involve local processing only. • **On 401 Status Code** The BIG-IP system forwards the user's HTTP request to the web server first without inserting a new Authorization header; (any Authorization header from a browser is also deleted). If the server requests authentication by responding with a 401 status code, the BIG-IP system retries the request with the Authorization header. The Kerberos ticket GSSAPI representation uses the SPNEGO mechanism displays (OID 1.3.6.1.5.5.2). Selecting **On 401 Status Code** results in an additional BIG-IP system and server request round trip when authentication is required for the request.

Setting	Value	Additional Information
Username Conversion	Check box is cleared by default.	When the check box is selected, the PREWIN2k/UPN user name input format is converted to the format you want to use for SSO. For example, convert `domain\username` or `username@domain` to `username`.

Kerberos SSO session variable list

The following session variables are used by Kerberos SSO.

Session Variable name	Description
session.logon.last.domain	Contains the user's Kerberos realm. If unset, the user's realm is the same as the server's realm. The variable name is specified as *User Realm Source* in the SSO configuration and can be changed.
session.logon.last.username	Contains the user's login name. This can be extracted from the client certificate or supplied by the user on the login screen. The variable name is specified as *UsernameSource* in the SSO configuration and can be changed.
session.logon.last.username.sso.state	This is set to `1` internally when Kerberos SSO fails. When this variable is set, all subsequent requests are passed to the application server without applying SSO for the remainder of the user session. The variable name is constructed by appending *.sso.state* to the name specified in **Username Source**.

Tips for successfully deploying Kerberos SSO

If you run into problems with Kerberos SSO, follow these tips to try to resolve issues.

Microsoft® IIS servers

Only Microsoft® IIS servers are supported for pool members or web application resources. First, make sure the server computers running IIS are members of your AD Domain. Then follow these steps to enable Kerberos in IIS Manager:

1. From Active Directory administrative tool, right-click **Web Sites** and select **Properties**.
2. Select the Directory Security tab and in the Authentication and Access Control area click **Edit**.
3. Clear **Enable Anonymous Access**.
4. Check **Integrated Windows Authentication** and click **OK**. You might need to restart IIS or reboot the server for this to take effect.

Reverse DNS resolution

Kerberos SSO relies on reverse DNS resolution for determining the SPN (Service Principal Name) for each server host, such as a load balanced pool member or a web application resource host. Access Policy Manager® should be configured to use DNS servers that have the appropriate forward and reverse DNS records for those servers. If DNS is lacking, those record host entries can be configured on the BIG-IP® system.

DNS and KDC

Kerberos SSO relies on DNS for KDC discovery when KDC is not specified in an SSO configuration. The DNS server should have SRV records pointing to the KDC servers for the realm's domain. When DNS is not properly configured, or if the realm's DNS domain name is different from the realm's name, you can specify the KDC by adding a realm section to `/etc/krb5.conf` file on the BIG-IP system. For DNS discovery to work, the `dns_lookup_kdc` option in the `[libdefaults]` section of that file must be set to `true`.

Credential Caching

Kerberos uses credential caching to store Kerberos tickets. Access Policy Manager uses the websso process to maintain credential caches in memory, so restarting the websso process will discard all Kerberos tickets used for SSO.

Credential caching and high availability

The Ticket cache is not synchronized between units in high availability. Each user's Kerberos tickets are stored in a separate cache, where the name is constructed from the username, the user Kerberos realm, and the server Kerberos realm. Each cache contains a copy of the delegation account TGT for the server realm, a S4U2Self ticket for the user for the server realm, and multiple S4U2Proxy tickets for the servers. Once all tickets are fetched and stored in the cache, they remain there until they expire according to their lifetime. Processing of subsequent SSO requests should not require any more queries to the KDC. Since all tickets obtained from the same TGT have that TGT's lifetime, all tickets in the cache expire simultaneously. Each user's cache exists independently from the user's session. If the user has multiple concurrent or sequential sessions, the sessions all share the same cache, as long as it remains valid. The cache continues to exist even without any active sessions.

Maximum number of cache entries

The maximum number of cache entries is set to 20000. If the number is exceeded, it destroys older entries using the LRU algorithm. Delegation account TGT for each server realm is fetched when the first user request for that realm is processed. The TGT is cached and copied into every user's cache when the user accesses servers in that realm. If the TGT remaining lifetime becomes more than one hour shorter than the configured lifetime, the TGT is re-fetched. This is done to ensure that the new user's tickets are fetched with the initial lifetime closer to the configured value, and to avoid all tickets expiring at the same time, causing a performance impact.

Single Sign-On and Multi-Domain Support

About multi-domain support for SSO

Access Policy Manager® (APM®) provides a method to enable users to use a single login or session across multiple virtual servers in separate domains. Users can access back-end applications through multiple domains or through multiple hosts within a single domain, eliminating additional credential requests when they go through those multiple domains. With multi-domain support, you have the option of applying different SSO methods across different domains.

***Important:** To enable multi-domain support, all virtual servers must be on a single BIG-IP® system.*

These are some of the benefits that APM provides when you use it to set up multi-domain support for SSO.

- Users can sign out from all domains at once.
- Users can move from one domain to another seamlessly. This eliminates the need re-run the access policy, and thus maintains the established session for the user.
- Administrators can configure different cookie settings (Secure, Host/Domain and Persistent) for different domains, and for different hosts within same domain
- Administrators can set up multiple SSO configurations to sign users in to multiple back-end applications for a single APM® session

How does multi-domain support work for SSO?

The configuration process in which you successfully set up multi-domain support for SSO requires the following elements.

- An access profile that includes a set of participating domains.
- An SSO configuration associated with each of the domains. Additionally, a designated URL that specifies the primary authentication service is included in the access profile.

  ***Note:** The host name of the URL is a virtual server that provides an access policy to retrieve the credentials from the user. If an un-authenticated user reaches any domain specified in the domain group, a re-direct is first made to the primary authenticating service so that credentials are collected in order to establish a session.*
- A virtual server.
- The access profile associated with each of the virtual servers participating in the domain group.

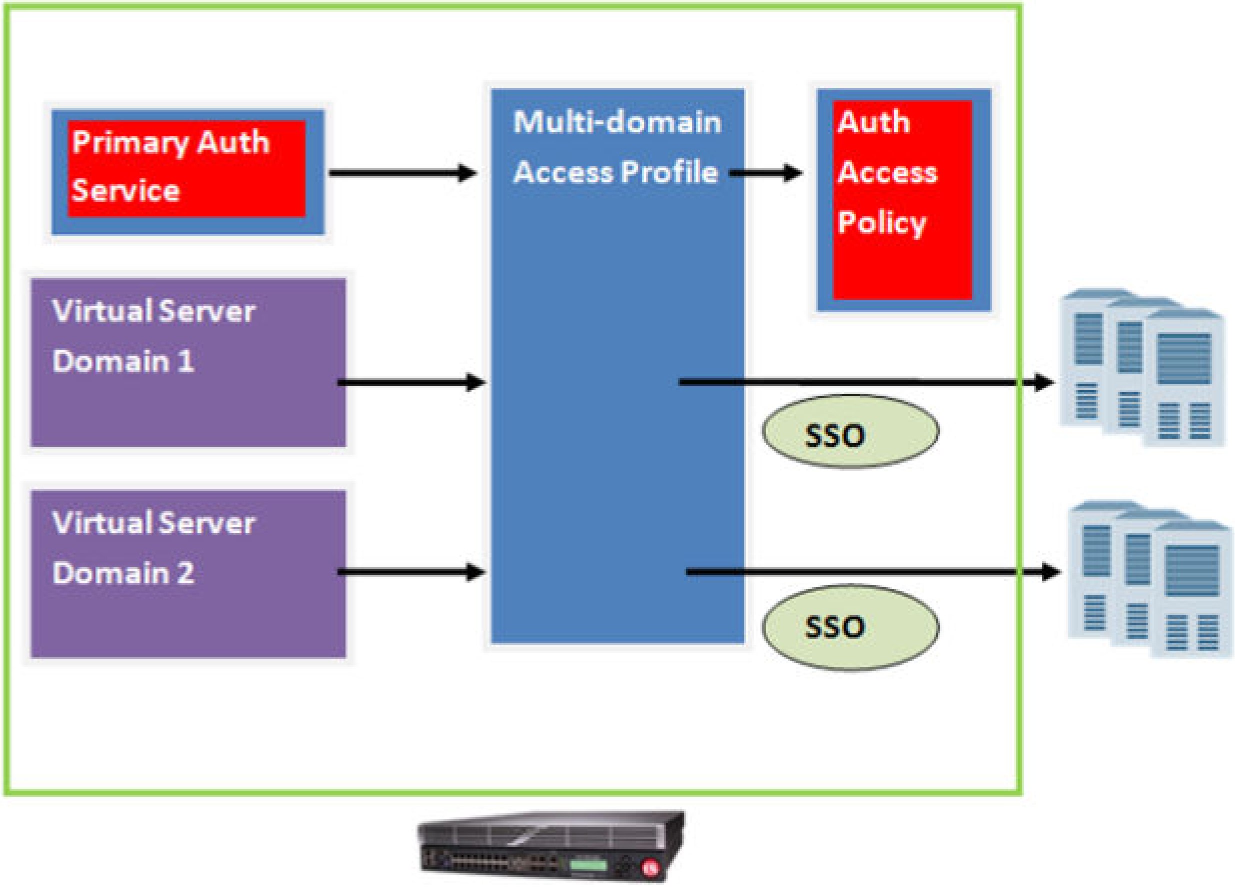


Figure 23: Configuration process for multi-domain support for SSO

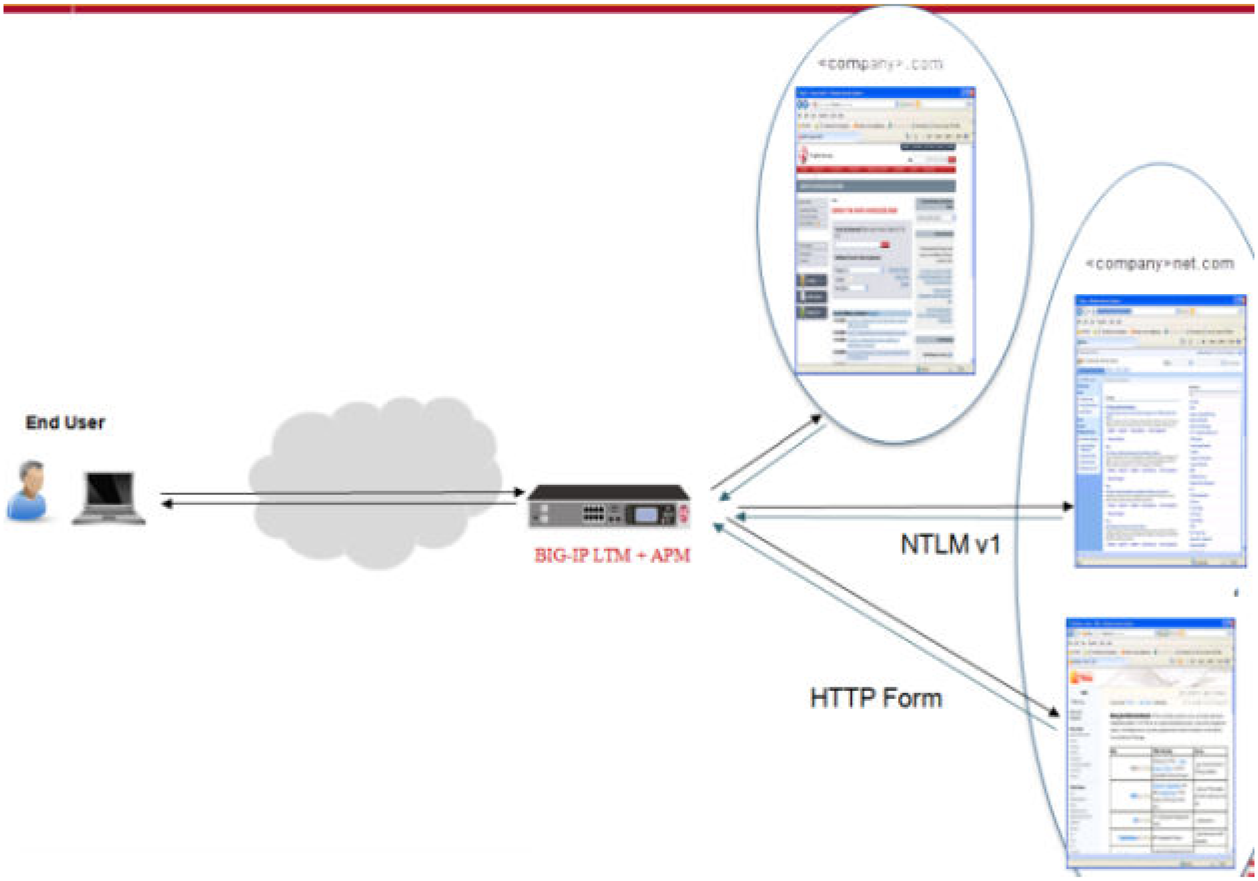


Figure 24: How multi-domain support for SSO works

Task summary for configuring domain support for SSO

Access Policy Manager® SSO lets you configure either a single domain or multiple domains for SSO.

To set up this configuration, follow the procedures in the task list.

Task List

Configuring an access policy for SSO single domain support
Configuring an access policy for SSO multi-domain support
Verifying log settings for the access profile
Creating a virtual server for SSO multi-domain support

Configuring an access policy for SSO single domain support

These steps apply only if you are setting up your access policy for SSO single domain support.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. From the list, select an access profile in which you want to add SSO capability.
   The properties screen for that access profile opens.
3. On the menu bar, click **SSO/Auth Domains**.
4. For **Domain Mode**, select **Single Domain**.

5. For the **SSO Configuration** setting, select an available SSO configuration from the list to apply to your access policy.
6. Click **Update**.
7. On the menu bar, click **Access Policy**.
8. Click the name of the access profile for which you want to edit the access policy.
   The properties screen opens for the profile you want to edit.
9. In the General Properties area, click the **Edit Access Policy for Profile** ***`profile_name`*** link.
   The visual policy editor opens the access policy in a separate screen.
10. Click the **(+)** icon anywhere in the access policy to add a new item.

    Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

    A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
11. For Predefined Actions, under General Purpose, select **SSO Credential Mapping**, and click **Add item**.
12. Click **Save**.
    You have now added SSO capability to your access policy.

Configuring an access policy for SSO multi-domain support

A user should be able to connect to any one of the virtual servers that participate in the domain group, and receive a request for credentials only once. Subsequent connections to other virtual servers within the domain group should not require the users to provide their credentials.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. From the list, select an access profile to which you want to add SSO capability.
   The properties screen for that access profile opens.
3. On the menu bar, click **SSO/Auth Domains**.
4. For the **Domain Mode** setting, select **Multiple Domains**.
5. For **Primary Authentication URI**, type the URI the client is directed to, for example, `http://login.com`, in order to receive an Access Policy Manager® session.

   Each domain that you configure indicates the domain the Access Policy Manager session (established by the primary authentication URI) is bound to.
6. In the Authentication Domain Configuration area, configure the **Cookie** setting by selecting **Host** or **Domain**, and typing the IP address for the host or, for domain, typing the fully qualified domain name.
7. Select **Cookie Options**. By default, **Secure** is selected.
8. From the **SSO Configuration** list, select the configuration that you want to associate to each host or domain. (Defaults to **None**.)
9. Click **Update**.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: ***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Creating a virtual server for SSO multi-domain support

For every domain, a virtual server should be configured.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address in CIDR format.

   The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is `10.0.0.1` or `10.0.0.0/24`, and an IPv6 address/prefix is `ffe1::0020/64` or `2001:ed8:77b5:2:10:10:100:42/64`. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a `/32` prefix.

   ***Note:** The IP address for this field needs to be on the same subnet as the external self-IP address.*
5. From **Access Profile**, select the profile you wish to attach to the virtual server.
6. Click **Finished**.

These steps should be repeated for every domain you specify in your access policy.

Common Deployment Examples for Single Sign-On

Common use cases for Single Sign-On deployment

You can deploy Single Sign-On in a variety of ways, depending on the needs within your networking environment. Deployment options include the following choices.

Use case deployment type	Description
For local traffic pool members	Deploy SSO for local traffic with pool members. The Web Application Access Management for Local Traffic Virtual Servers wizard can be used for this deployment.
For web application access over network access	Deploy SSO through a network access tunnel with matching virtual servers enabled on the connectivity interface.
For web applications	Deploy SSO so users can access their web applications. You can assign an SSO object as part of the web application resource item, such as a SAML resource or a portal acess resource item, or assign the object at the access profile level instead.

Overview: Configuring SSO for web apps over network access

Without implementing single-sign on (SSO) for web applications, remote clients that try to access web services over a network access connection must supply credentials multiple times.

This implementation to support SSO includes a typical network access configuration with a secure connectivity (tunnel) interface. Additional configuration to support SSO is required for each web service.

The configuration for each web service includes a virtual server that is enabled on the tunnel and that specifies a destination address to match the web server. An SSO access profile type is required on the virtual server. An *SSO access profile type* specifies an SSO configuration; no access policy is associated with this profile type.

It is possible for a matching virtual server for a web application to match a resource specified in a portal access resource item. (Although not required, portal access resources can be assigned to the webtop in the network access configuration.) In this case, SSO configuration must be specified at the access profile level (in the virtual server) and not in the portal access resource item.

Task summary

Configuring a network access resource
Configuring network access properties
Creating a connectivity profile
Creating an access profile for remote access
Verifying log settings for the access profile
Adding network access to an access policy
Configuring a virtual server for network access
Creating an SSO configuration

Creating an access profile for web app SSO
Configuring a virtual server for web app SSO

Configuring a network access resource

Configure a network access resource to provide secure access to corporate applications and data using a standard web browser, or the BIG-IP Edge Client®.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Network Access (VPN)** > **Network Access Lists**.
   The Network Access Lists screen opens.
2. Click the **Create** button.
   The New Resource screen opens.
3. In the **Name** field, type a name for the resource.
4. To automatically start this network access resource when a client reaches a webtop to which the resource is assigned, select the **Auto launch** check box.

   Note: *When multiple network access resources are assigned to a webtop, Auto launch can be enabled for only one network access resource.*
5. In the Customization Settings for English area, in the **Caption** field, type a caption.

   The caption appears on the full webtop, and is required.
6. Click the **Finished** button.
   The Network Access configuration screen opens, and you can configure the properties for the network access resource.

Configuring network access properties

Configure properties for a network access resource to specify network settings and the optimized applications, hosts, drives, and applications that a remote user can access through the network access resource.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Network Access (VPN)** > **Network Access Lists**.
   The Network Access Lists screen opens.
2. Click the name to select a network access resource on the Resource List.
   The Network Access editing screen opens.
3. To configure the network settings for the network access resource, click **Network Settings** on the menu bar.
4. To configure DNS and hosts settings for the network access resource, click **DNS/Hosts** on the menu bar.
5. To configure the drive mappings for the network access resource, click **Drive Mappings** on the menu bar.
6. To configure applications to start for clients that establish a Network Access connection with this resource, click **Launch Applications** on the menu bar.

Creating a connectivity profile

You create a connectivity profile to configure client connections.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Connectivity** > **Profiles**.
   A list of connectivity profiles displays.
2. Click **Add**.

The Create New Connectivity Profile popup screen opens and displays General Settings.

3. Type a **Profile Name** for the connectivity profile.
4. Select a **Parent Profile** from the list.

   APM® provides a default profile, **connectivity**.
5. Click **OK**.
   The popup screen closes, and the Connectivity Profile List displays.

The connectivity profile displays in the list.

To provide functionality with a connectivity profile, you must add the connectivity profile and an access profile to a virtual server.

Creating an access profile for remote access

You create an access profile to specify any access policy configuration for a virtual server that serves network access, portal access, or application access traffic.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *An access profile name must be unique among all per-session profile and per-request policy names.*
4. From the **Profile Type** list, select **SSL-VPN**.

   Selecting this profile type restricts the access policy items displayed in the visual policy editor to those that contribute to a correct remote access configuration.

   Additional fields display set to default values.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

   This creates an access profile with a default access policy.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.

4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding network access to an access policy

Before you assign a network access resource to an access policy, you must define a network access webtop or a full webtop.

When you assign a network access resource to an access policy branch, a client that successfully completes the branch rule, starts a network access tunnel.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile for which you want to edit the access policy.
   The properties screen opens for the profile you want to edit.
3. On the menu bar, click **Access Policy**.
4. In the General Properties area, click the **Edit Access Policy for Profile** ***`profile_name`*** link.
   The visual policy editor opens the access policy in a separate screen.
5. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
6. Select **Advanced Resource Assign** and click **Add**.
7. Select the resources to add:
   a) On the Resource Assignment screen, click **Add New Entry**, then click **Add/Delete**
   b) On the Webtop tab, select one full or network access webtop.
   c) On the Network tab, select one or more network access resources.
   d) If you assigned a full webtop, select any other types of resources that you want to add.
   e) Click **Update**.

      If you add a full webtop and multiple network access resources, Auto launch can be enabled for only one network access resource. (With Auto launch enabled, a network access resource starts automatically when the user reaches the webtop.)
8. Click **Save**.
9. Click **Apply Access Policy** to save your configuration.

A network access tunnel and a webtop are assigned to the access policy. On a full webtop, a user can click the Network Access link to start a network access tunnel; or, if one network access tunnel is configured with Auto launch enabled, the tunnel can start automatically.

Configuring a virtual server for network access

Create a virtual server to which the network access associates your access policy.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address in CIDR format.

   The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is `10.0.0.1` or `10.0.0.0/24`, and an IPv6 address/prefix is `ffe1::0020/64` or `2001:ed8:77b5:2:10:10:100:42/64`. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a `/32` prefix.

   ***Note:** The IP address you type must be available and not in the loopback network.*
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. From the **Configuration** list, select **Advanced**.
7. From the **HTTP Profile** list, select **http**.
8. In the Configuration area, specify both **SSL Profile (Client)** and **SSL Profile (Server).**
9. From the **Source Address Translation** list, select **Auto Map**.
10. In the Access Policy area, select the **Access Profile** you created for remote access.
11. From the **Connectivity Profile** list, select the connectivity profile.
12. Click **Finished**.

Creating an SSO configuration

Creating an SSO configuration is a necessary first step for supporting single sign-on.

***Note:** Access Policy Manager® (APM®) supports several types of SSO configuration. Refer to BIG-IP®Access Policy Manager®: Authentication and Single Sign-on in the AskF5™ Knowledge Base at `http://support.f5.com/kb/en-us.html`.*

1. On the Main tab, select **Access** > **Single Sign-On**.
   The Single Sign-On screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. From the SSO Configurations by Type menu, choose an SSO type.
   A screen appears, displaying SSO configurations of the type you specified.
4. In the **Name** field, type a name for the SSO configuration.
5. Specify all relevant parameters.
6. Click **Finished**.

Creating an access profile for web app SSO

Before you start, you must create an SSO configuration for the web application for which you want to support single sign-on.

Configure an access profile of type SSO to provide single sign-on over a network access tunnel for a web application.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.

3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select **SSO**.
5. From the **SSO Configuration** list, select the configuration that you created for the web application.
6. Click **Finished**.

This creates an access profile for which there is no access policy.

Configuring a virtual server for web app SSO

For each web application, you must have previously created a virtual server with a destination address that matches that of the web server.

Configure settings on the virtual server for each web service that clients access over the network tunnel to eliminate the need for clients to enter credentials multiple times.

Note: *The name of the secure connectivity interface on which this virtual server must be enabled is the name of the connectivity profile specified for the virtual server for network access.*

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Select the virtual server that was previously created for the web service.
   The General Properties screen opens.
3. Scroll down to the **VLAN and Tunnel Traffic** setting and select **Enabled on**.
4. For the **VLANs and Tunnels** setting, move the secure connectivity interface to the **Selected** list.
5. From the Configuration list, select **Advanced**, scroll down, and make sure that the **Address Translation** and **Port Translation** check boxes are cleared.
6. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
7. Click **Update**.
   The users are now able to access this web service without entering credentials multiple times.

About SSO for portal access resources

An SSO configuration can be specified in a portal access resource item or in the access profile through which the portal access resource is assigned in the access policy.

If a portal access resource item and a virtual server that matches the resource populate the same session, an SSO configuration must be specified only once and at the access profile level. The SSO configuration must be specified in the access profile for the matching virtual server and not in the portal access resource item.

Configuring SSO for a portal access resource item

You must have created a portal access resource and added one or more resource items to it. You must have created an SSO configuration.

Add an SSO configuration to a portal access resource item to support SSO at the resource level instead of supporting SSO at the access profile level.

1. On the Main tab, click **Access** > **Connectivity / VPN** > **Portal Access** > **Portal Access Lists**.

The Portal Access List screen opens.

2. In the **Resource Items** column, click the link for a resource item.
   A Properties screen for that resource item opens.
3. In the **Resource Item Properties** area from the **SSO Configuration** list, select an SSO configuration.

   The default value is **None**.
4. Click **Update**.
   The Properties screen refreshes.

To add SSO configurations to additional portal access resource items, repeat these steps.

Introducing Access Policy Manager SAML Support

About SAML

Security Assertion Markup Language (SAML) defines a common XML framework for creating, requesting, and exchanging authentication and authorization data among entities known as Identity Providers (IdPs) and Service Providers (SPs). This exchange enables single sign-on among such entities.

- *IdP* is a system or administrative domain that asserts information about a subject. The information that an IdP asserts pertains to authentication, attributes, and authorization. An assertion is a claim that an IdP makes about a subject.
- *Service Provider* is a system or administrative domain that relies on information provided by an IdP. Based on an assertion from an IdP, a service provider grants or denies access to protected services.

In simple terms, an IdP is a claims producer and a service provider is a claims consumer. An IdP produces assertions about users, attesting to their identities. Service providers consume and validate assertions before providing access to resources.

SAML 2.0 is an OASIS open standard. The SAML core specification defines the structure and content of assertions. For the SAML 2.0 features that (Access Policy Manager®) (APM®) supports, see solution article *sol16497* on the AskF5™ web site located at `http://support.f5.com/`.

About SAML metadata

SAML metadata specifies how configuration information is defined and shared between two communicating entities: a SAML Identity Provider (IdP) and a SAML service provider. Service provider metadata provides information about service provider requirements, such as whether the service provider requires a signed assertion, the protocol binding support for endpoints (AssertionConsumerService) and which certificates and keys to use for signing and encryption. IdP metadata provides information about IdP requirements, such as the protocol binding support for endpoints (SingleSignOnService), and which certificate to use for signing and encryption.

About SAML single logout service

Single logout (SLO) service is a way to allow a user to terminate all sessions in an automatic manner without user intervention. A SAML Identity Provider (IdP) or the SAML service provider (SP) can initiate logout. The SAML IdP coordinates all logouts. When a SAML SP initiates a logout it contacts the SAML IdP to carry out the coordinated logout on its behalf.

Access Policy Manager® (APM®) supports SLO when all participating entities (SAML SPs and IdPs) support SLO. APM supports HTTP-POST binding for SLO messages.

About SAML artifact resolution protocol

SAML artifact resolution protocol provides a mechanism by which a service provider (SP) can obtain a SAML assertion from an Identity Provider (IdP) by reference. Instead of binding an assertion to a transport protocol, an IdP sends a small piece of data (known as an artifact) using either HTTP POST or HTTP Redirect bindings. An SP can then use artifact resolution protocol with the SOAP binding protocol to resolve the artifact into the original assertion.

Although the SAML 2.0 specification supports using an artifact in place of any SAML message (request or response), the BIG-IP® system supports using artifacts for assertions only.

When BIG-IP is configured as a SAML IdP, an artifact resolution service on the BIG-IP system can process artifact resolution requests and artifact resolution responses.

When BIG-IP is configured as a SAML SP, it can send the artifacts it receives to a URL that the IdP specifies for resolving artifacts into assertions.

About the benefits of using APM for SAML support

Access Policy Manager as a SAML Identity Provider (IdP)

When you use Access Policy Manager®(APM®) as a SAML IdP, APM can authenticate and generate assertions for a user who can then gain access to resources protected by SAML. APM provides SAML assertions (claims) that service providers verify and consume. In this role, APM acts as an authentication server and provides single sign-on to service provider resources.

Access Policy Manager as a SAML Service Provider (SP)

When you use APM as a SAML service provider, APM consumes SAML assertions (claims) and validates their trustworthiness. After successfully verifying the assertion, APM creates session variables from the assertion contents. In an access policy, you can use these session variables to finely control access to resources and to determine which ACLs to assign. Based on the values of session variables, you can create multiple branches in the policy, assigning different resources and different ACLs on each branch. When it runs, the access policy follows a branch depending on the values of session variables.

Federation

APM systems operate with one another when one APM system is configured as an IdP and other APM systems are configured as service providers. This allows a user to authenticate with one APM acting as an IdP, and then use any number of APM systems, serving as service providers, without having to re-authenticate.

Metadata import and export

You can simplify SAML configuration using metadata files. When you use APM as an IdP, you can configure a SAML service provider by importing a metadata file that you obtain from the vendor. Similarly, when you use APM as a service provider, you can configure an IdP by importing a metadata file that you obtain from the vendor. You can export the metadata for APM as a SAML IdP from APM and import the metadata file into a service provider (or use information from the metadata file to configure the service provider). You can export the metadata for APM as a SAML service provider from APM and import the metadata file into an IdP (or use information from the metadata file to configure the IdP).

Templates

APM provides a few templates that you can use to create service provider connectors, and a few that you can use to create IdP connectors with a minimal amount of typing.

Custom service providers and custom IdPs

In addition to configuring service provider connectors or an IdP connector from vendor metadata files or APM templates, you can configure custom service provider and IdP connectors.

IdP-initiated and service provider-initiated client connections

Access Policy Manager supports client connections that initiate at the IdP or at the service provider.

Signed assertions

By default, APM produces signed assertions. An assertion signed by the asserting party (the IdP) supports assertion integrity, authentication of the asserting party to a SAML relying party (a service provider), and, if the signature is based on the SAML authority's public-private key pair, non-repudiation of origin.

Encrypted assertions

For increased security, APM can optionally encrypt the entire assertion. APM supports encryption methods AES128, AES192, and AES256.

Support for SAML profiles

APM supports the Web Browser SSO profile with HTTP redirect and HTTP POST bindings. APM also supports Enhanced Client or Proxy Profile (ECP).

About support for Microsoft Office 365 as a SAML service provider

APM® supports Microsoft Office 365 as a SAML service provider (SP). The BIG-IP® system, configured as a SAML Identity Provider (IdP), supports the Enhanced Client or Proxy Profile (ECP) SAML profile. APM includes a predefined external service provider connector for Office 365. The SP connector supports assertion consumer services with PAOS (HTTP reverse SOAP) and POST bindings.

When should I configure a BIG-IP system as a SAML IdP?

Configure a BIG-IP® system as a SAML identity provider (IdP) when you have one BIG-IP system and you want it to provide single sign-on authentication service for a group of external SAML service providers.

When should I configure a BIG-IP system as a SAML service provider?

Configure a BIG-IP® system as a SAML service provider when you have one BIG-IP system and you want it to protect services that are behind it, and direct users to an external SAML identity provider for authentication.

Overview: Exchanging certificates among SAML entities

For security purposes, each SAML service provider (SP) should have a certificate from the SAML Identity Provider (IdP) that manages identities for it; each IdP should have certificates from the SPs for which it manages identities.

Certificates on the BIG-IP system

Metadata normally includes a certificate. When you import metadata into a BIG-IP® system from an external SP or an external IdP, the certificate that was included in the metadata is stored on the BIG-IP system. When you configure security-related settings on the BIG-IP system, you select certificates from the store.

If you do not have metadata that you can import from external SPs or IdPs, then you need to do one of the following:

- Get certificate files that you can import from the external systems into the BIG-IP system.
- Get certificate information from each external system that you can then paste into a user interface to create certificate files for them on the BIG-IP system.

BIG-IP system certificates on external systems

To get a certificate from the BIG-IP system, you can export it. You can potentially also get a certificate from a BIG-IP system by exporting SAML metadata for use on the external system.

When you export metadata from a BIG-IP system, it includes a certificate. However, when an external system requires signed metadata, the external system must already have a certificate from the BIG-IP system to validate the metadata.

Task Summary

Importing an SSL certificate

Exporting a digital certificate

Importing an SSL certificate

Before you can perform this procedure, an SSL certificate must be available.

A BIG-IP® system requires a certificate from an external SAML service provider (SP) when the BIG-IP system is configured as a SAML Identity Provider (IdP) and must verify a signed authentication request from the SP. A BIG-IP system requires a certificate from an external IdP when the BIG-IP system is configured as an SP and must verify a signed authentication request from the IdP.

1. On the Main tab, click **System** > **Certificate Management** > **Traffic Certificate Management**.
   The Traffic Certificate Management screen opens.
2. Click the **Import** button.
3. From the **Import Type** list, select **Certificate**.
4. For the **Certificate Name** setting:
   - If you are importing a new certificate, select **Create New** and type a unique name in the field.
   - If you are replacing an existing certificate, select **Overwrite Existing** and select a certificate name from the list.
5. For the **Certificate Source** setting, select **Upload File** and browse to select the certificate you obtained from the vendor.
6. Click **Import**.

The SSL certificate for the vendor is installed.

Exporting a digital certificate

You export a digital certificate when you configure a BIG-IP® system for SAML and you need a certificate from the BIG-IP system on an external SAML system.

1. On the Main tab, click **System** > **Certificate Management** > **Traffic Certificate Management**.
   The Traffic Certificate Management screen opens.
2. Click the name of the certificate you want to export.
   The General Properties screen displays.
3. Click **Export**.
   The Certificate Export screen displays the contents of the certificate in the **Certificate Text** box.
4. To obtain the certificate, do one of the following:
   - Copy the text from the **Certificate Text** field, and paste it as needed into an interface on another system.
   - At the **Certificate File** option, click **Download filename** where the filename is the name of the certificate file, such as `mycert.crt`.

Using APM as a SAML IdP (SSO portal)

Overview: Configuring BIG-IP as IdP for IdP- and SP-initiated connections

This configuration supports:

- An SSO portal on the BIG-IP® system configured as a SAML Identity Provider (IdP).
- Service providers (SPs) with the same or different requirements for assertion type and value and attributes (provided by the IdP).
- SP- and IdP-initiated connections.

About local IdP service

A *SAML IdP service* is a type of single sign-on (SSO) authentication service in Access Policy Manager® (APM®). When you use a BIG-IP® system as a SAML identity provider (IdP), a SAML IdP service provides SSO authentication for external SAML service providers (SPs). You must bind a SAML IdP service to SAML SP connectors, each of which specifies an external SP. APM responds to authentication requests from the service providers and produces assertions for them.

About SP connectors

A SAML service provider connector (an SP connector) specifies how a BIG-IP® system, configured as a SAML Identity Provider (IdP), connects with an external service provider.

What are the available ways I can configure a SAML SP connector?

You can use one or more of these methods to configure SAML service provider (SP) connectors in Access Policy Manager®.

- From metadata - Obtain a metadata file from the vendor and import it into Access Policy Manager. The advantage to this method is that the vendor provides the majority of all required data, including certificates. You can complete the configuration by simply typing a unique name for the SP connector, a very few additional required fields, and browsing to and importing the file. Access Policy Manager then configures the SP connector.
- From template - Use templates that Access Policy Manager provides for some vendors; for example, Google. The advantages to this method are that:

  - Most required data is included in the template
  - Additional required data is minimal. You can obtain it and certificates from the vendor

  After you select a template and type data into a few fields, Access Policy Manager configures the SP connector.
- Custom - Obtain information from the vendor and type the settings into the Configuration utility. To use this method, you must also obtain certificates from the vendor and import them into the BIG-IP® system. Use this method when a metadata file or a template for an SP connector is not available.

Task summary

Setting up a BIG-IP® system as a SAML identity provider (IdP) system involves two major activities:

- First, you set up connection from the BIG-IP system to the external SAML service providers (SPs)

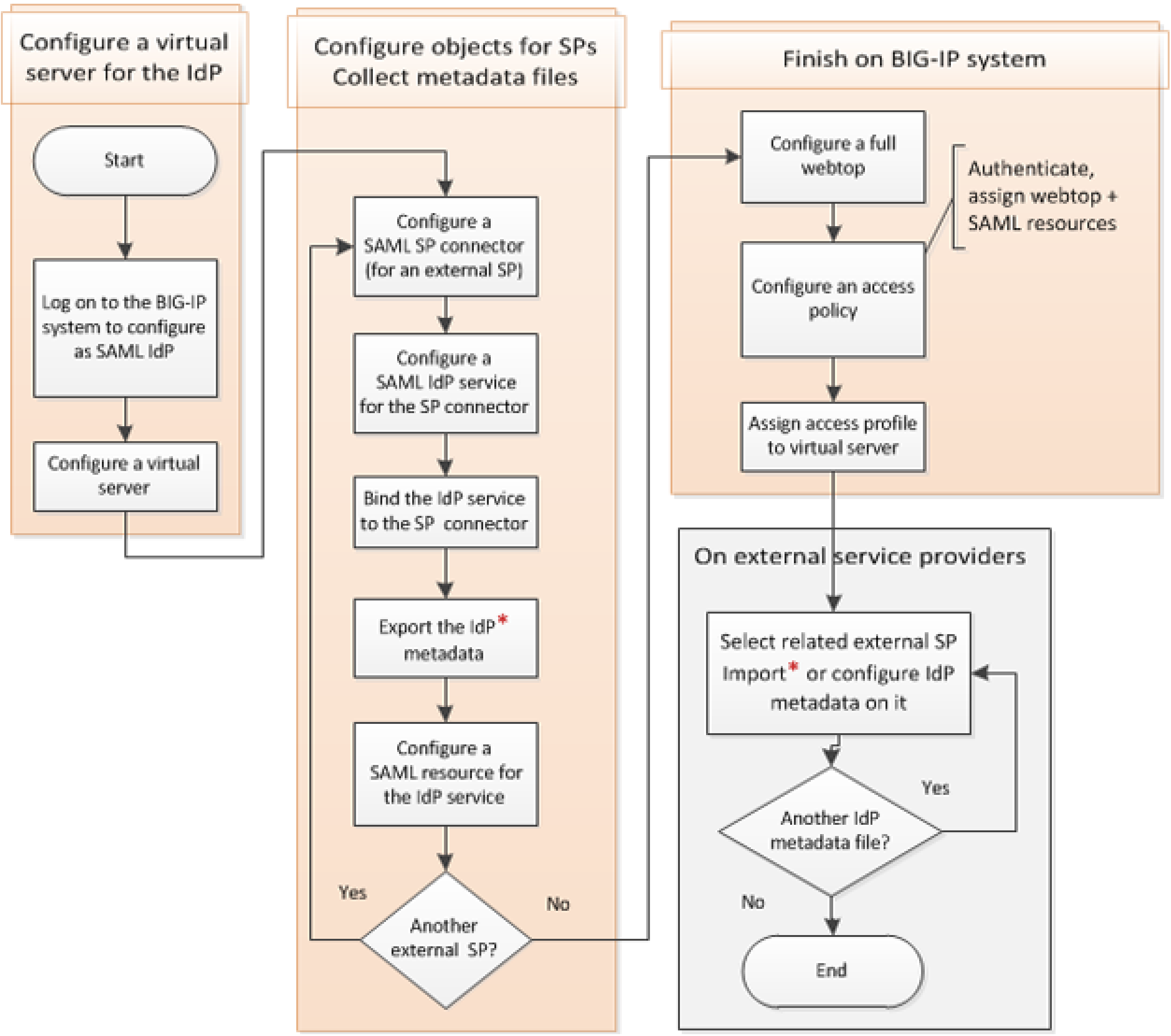
- Then, you set up connection from the external SAML SPs to the BIG-IP system

Task list

Creating a virtual server for a BIG-IP (as SAML IdP) system
Configuring an artifact resolution service
Configuring SAML SP connectors
Configuring a full webtop
Configuring an access policy for a SAML SSO portal
Verifying log settings for the access profile
Adding the access profile to the virtual server
Adding IdP metadata from APM to external SAML SPs

Flowchart: Configuration to support a SAML SSO portal

This flowchart illustrates the process for configuring a BIG-IP® system as a SAML identity provider (IdP) that provides an SSO portal.


Configure a virtual server for the IdP
Start
Log on to the BIG-IP system to configure as SAML IdP
Configure a virtual server
Configure objects for SPs Collect metadata files
Configure a SAML SP connector (for an external SP)
Configure a SAML IdP service for the SP connector
Bind the IdP service to the SP connector
Export the IdP* metadata
Configure a SAML resource for the IdP service
Yes
Another external SP?
No
Finish on BIG-IP system
Configure a full webtop
Authenticate, assign webtop + SAML resources
Configure an access policy
Assign access profile to virtual server
On external service providers
Select related external SP Import* or configure IdP metadata on it
Yes
Another IdP metadata file?
No
End

Creating a virtual server for a BIG-IP (as SAML IdP) system

Before you start this task, configure a client SSL profile and a server SSL profile if you are going to create an SSL virtual server.

***Note:** Access Policy Manager® supports using a non-SSL virtual server for the BIG-IP® system configured as a SAML Identity Provider (IdP). However, we recommend using an SSL virtual server for security reasons. The following procedures include steps that are required for configuring an SSL virtual server, such as selecting client and server SSL profiles, and setting the service port to HTTPS.*

Specify a host virtual server to use as the SAML IdP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an IdP now appears on the Virtual Server List. The virtual server destination is available for use in one or more SAML IdP service configurations.

Configuring an artifact resolution service

Before you configure the artifact resolution service (ARS), you need to have configured a virtual server. That virtual server can be the same as the one used for the SAML Identity Provider (IdP), or you can create an additional virtual server.

***Note:** F5® highly recommends that the virtual server definition include a server SSL profile.*

You configure an ARS so that a BIG-IP® system that is configured as a SAML IdP can provide SAML artifacts in place of assertions. With ARS, the BIG-IP system can receive Artifact Resolve Requests (ARRQ) from service providers, and provide Artifact Resolve Responses (ARRP) for them.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **Artifact Resolution Services**.
2. Click **Create**.
   The Create New SAML Artifact Resolution Service popup screen opens, showing general settings.
3. In the **Name** field, type a name for the artifact resolution service.
4. In the **Description** field, type a new description.
5. Click **Service Settings.**
6. From the **Virtual Server** list, select the virtual server that you created previously.

   ARS listens on the IP address and port configured on the virtual server.

7. In the **Artifact Validity (Seconds)** field, type the number of seconds for which the artifact remains valid. The default is 60 seconds.

   The BIG-IP® system deletes the artifact if the number of seconds exceeds the artifact validity number.
8. For the **Send Method** setting, select the binding to use to send the artifact, either **POST** or **Redirect**.
9. In the **Host** field, type the host name defined for the virtual server, for example **ars.siterequest.com**.
10. In the **Port** field, type the port number defined in the virtual server. The default is `443`.
11. Click **Security Settings.**
12. To require that artifact resolution messages from an SP be signed, select the **Sign Artifact Resolution Request** check box.
13. To use HTTP Basic authentication for artifact resolution request messages, in the **User Name** field, type a name for the artifact resolution service request and in the **Password** field, type a password.

    These credentials must be present in all Artifact Resolve Requests sent to this ARS.
14. Click **OK**.
    The popup screen closes, leaving the Artifact Resolution Services list screen open.

The Artifact Resolution Service is ready for use.

Configuring SAML SP connectors

Before you can configure a SAML service provider, you must first obtain an SSL certificate from the SAML service provider (SP) and import it into the certificate store on the BIG-IP® system.

You configure information about a SAML service provider so that Access Policy Manager® (APM®) can act as a SAML Identity Provider (IdP) for it.

***Note:** Configure one SAML SP connector for each external SAML service provider for which this BIG-IP system provides SSO authentication service.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **External SP Connectors**.
   A list of SAML SP connectors displays.
2. Click **Create**.
   The Create New SAML SP Connector screen opens.
3. In the **Service Provider Name** field, type a unique name for the SAML SP connector.
4. In the **SP Entity ID** field, type a unique identifier for the service provider.

   This is usually a unique URI that represents the service provider. You should obtain this value from the service provider.
5. From the left pane, select **Endpoint Settings**.

   The appropriate settings are displayed.
6. (Optional) In the **Relay State** field, type a value.

   The relay state can be an absolute path, such as `/hr/index.html`; it can be a URL, such as `https://www.abc.com/index.html`; or, it can be anything that the service provider understands. The information passed in relay state could be used by the service provider according to business logic. For example, some service providers use relay state to maintain a session state, while others use it to perform an action, such as redirecting the user to the page passed in relay state. APM sends the relay state value back to the service provider as part of the assertion response in the `RelayState` parameter.

   When the `RelayState` parameter is already part of the authentication request to the BIG-IP system, APM returns the value that was sent in the request. Otherwise, APM uses the value from this configuration.
7. In the Assertion Consumer Services area, specify at least one assertion consumer service.

A service provider can use multiple bindings to receive an assertion from the Identity Provider. The service provider can specify a different assertion consumer service (ACS) URL for each binding, and provide a unique ACS URL index for the binding.

To support SAML artifacts, make sure that at least one ACS specifies the artifact binding.

a) Click **Add**.
A new row displays in the table.
b) In the **Index** field, type the index number, zero (0) or greater.
c) If this is the default service, select the **Default** check box.

You must specify one of the services as the default.

d) In the **Location URL** field, type the URL where the IdP can send an assertion to this service provider.

APM supports HTTP-Artifact binding, POAS (HTTP reverse SOAP) binding, and HTTP-POST binding to this service.

e) From the **Binding** list, select **Artifact**, **PAOS**, or **POST**.
f) Click **Update**.

8. From the left pane, select **Security Settings**.

a) If the SP should sign the authentication or the artifact resolution requests that it sends to the SAML IdP (this BIG-IP system), select the **Require Signed Authentication Request** check box, select a private key from the **Message Signing Private Key** list, and select a certificate from the **Message Signing Certificate** list.

This device (BIG-IP system as IdP) uses the certificate to verify the signature of the request from the SP.

b) To require that the SAML IdP sign the assertion before sending it to the SP, select the **Assertion must be signed** check box, and select an algorithm from the **Signing Algorithm** list.

Assertion must be signed is selected by default. Clearing this check box is not recommended.

c) To require that the SAML IdP sign the response before sending it to the SP, select the **Response must be signed** check box.

Note: *The algorithm specified in the* ***Signing Algorithm*** *list applies to a signed assertion and a signed response.*

d) To require that the SAML IdP encrypt the assertion before sending it to the SP, select the **Assertion must be encrypted** check box, select a type from the **Encryption Type** list, and select a certificate from the **Encryption Certificate** list.

APM supports AES128, AES192, and AES256 encryption types.

9. From the left pane, select **SLO Service Settings**.

SLO stands for Single Logout.

a) (Optional) In the **Single Logout Request URL** field, type a URL specifying where APM should send a logout request to this service provider when the BIG-IP system initiates a logout request.
b) In the **Single Logout Response URL** field, type a URL to which the SP should send a logout response for the BIG-IP system to indicate that single logout is complete.

Note: *APM supports HTTP-POST binding for the SLO service. For SLO to work, all entities (SPs and IdPs), must support SLO.*

10. Click **OK**.
The popup screen closes.

APM creates a SAML SP connector. It is available to bind to a SAML IdP service.

Configuring a SAML IdP service for one SP connector

Configure a SAML Identity Provider (IdP) service for Access Policy Manager®, as a SAML IdP, to provide single sign-on authentication for one SAML service provider (SP).

***Note:** Configure one IdP service for each SAML service provider.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Click **Create**.
   The Create New IdP Service popup screen displays.
3. In the **IdP Service Name** field, type a unique name for the SAML IdP service.
4. In the **IdP Entity ID** field, type a unique identifier for the IdP (this BIG-IP® system).

   Typically, the ID is a URI that points to the BIG-IP virtual server that is going to act as a SAML IdP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://siterequest.com/idp`, where the path points to the virtual server you use for BIG-IP system as a SAML IdP.
5. If the **IdP Entity ID** field does not contain a valid URI, you must provide one in the IdP Name Settings area:
   a) From the **Scheme** list select **https** or **http**.
   b) In the **Host** field, type a host name.
      For example, type `siterequest.com` in the **Host** field.
6. If you select **SAML Profiles** on the left pane, the **Web Browser SSO** check box is selected by default.

   At least one profile must be selected.
7. To specify that this IdP use an artifact resolution service, click **Endpoint Settings** on the left pane and select a service from the **Artifact Resolution Service** list.
8. On the left pane, select **Assertion Settings** and complete the settings that display:
   a) From the **Assertion Subject Type** list, select the type of subject for the IdP to authenticate.
   b) From the **Assertion Subject Value** list, select the name of a session variable.

      This variable, *`%{session.logon.last.username}`*, is generally applicable. Some session variables are applicable depending on the type of authentication that you use for your site.
   c) In the **Authentication Context Class Reference** field, select a URI reference.

      The URI reference identifies an authentication context class that describes an authentication context declaration.
   d) In the **Assertion Validity (in seconds)** field, type the number of seconds for which the assertion is valid.
   e) To encrypt the subject, select the **Enable encryption of Subject** check box.
      The **Encryption Strength** list becomes available.
   f) From the **Encryption Strength** list, select a value.

      Supported values are AES128, AES192, and AES256.
9. On the left pane, select **SAML Attributes**, and for each attribute that you want to include in the attribute statement, repeat these substeps.
   a) Click **Add**.
      A Create New SAML Attribute popup screen displays.
   b) In the **Name** field, type a unique name for the attribute.

      Usually, the name is a fixed string, but it can be a session variable.

c) To add a value to the attribute, click **Add**, type a value in the **Value(s)** field, and click **Update** to complete the addition.

   You can use a session variable for the value.

   This example shows using a fixed string for the name and a session variable for the value. Name: `user_telephonenumber` and value: `%{session.ad.last.attr.telephoneNumber}`.

   You can repeat this step to add multiple values for an attribute.

d) To encrypt the values, select the **Encrypt** check box and select a value from the **Type** list.

   Supported values for type are AES128, AES192, and AES256.

e) Click **OK**.
   The Create New SAML Attribute popup screen closes.

10. Click **Security Settings** from the left pane.
    a) From the **Signing Key** list, select the key from the BIG-IP system store.

       None is selected by default.

    b) From the **Signing Certificate** list, select the certificate from the BIG-IP system store.

       When selected, the IdP (the BIG-IP system) publishes this certificate to the service provider so the service provider can verify the assertion. **None** is selected by default.
11. Click **OK**.
    The popup screen closes. The new IdP service appears on the list.

Access Policy Manager® (APM®) creates a SAML IdP service. It is available to bind to an SP connector.

Binding a SAML IdP service to one SP connector

Bind a SAML Identity Provider (IdP) service and a SAML service provider (SP) connector so that the BIG-IP® system can provide authentication (SAML IdP service) to the external SAML service provider.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the list.

   Select an IdP service that you configured for use with one particular SP connector only.
3. Click **Bind/Unbind SP Connectors**.
   The screen displays a list of available SAML SP connectors.
4. Select the one SAML SP connector that you want to pair with this IdP service.
5. Select **OK**.
   The screen closes.

The SAML SP connector that you selected is bound to the SAML IdP service.

Exporting SAML IdP metadata from APM

You need to convey the SAML Identity Provider (IdP) metadata from Access Policy Manager® (APM®) to the external service provider that uses this SAML IdP service. Exporting the IdP metadata for a SAML IdP service to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the table and click **Export Metadata**.
   A popup screen opens, with **No** selected on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

      APM uses the key to sign the metadata.

c) From the **Signature Verification Certificate** list, select a certificate.

   APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.

4. Select **OK**.
   APM downloads an XML file.

You must either import the IdP metadata XML file on the service provider system or use the information in the file to configure the SAML IdP on the service provider system.

Configuring a SAML resource and attaching a SAML IdP service

Configure a SAML resource to provide access to services on a SAML service provider when using Access Policy Manager® (APM®) as a SAML IdP.

Note: *Configure one SAML resource for each SAML IdP service that you have configured.*

1. On the Main tab, click **Access** > **Federation** > **SAML Resources**.
   The SAML Resources list screen opens.
2. Click the **Create** button.
   The SAML Resource New Resource window opens
3. In the **Name** field, type a unique name for the SAML resource.
4. Do not clear the **Publish on Webtop** check box unless when you want to remove this resource from the webtop.

   When **Publish on Webtop** is selected, the SAML resource is displayed on a webtop where a user can initiate connection to an SP by clicking the icon. If you want users to initiate connection to this resource from an external SAML service provider only and do not want to show this resource on a webtop, clear the check box.
5. In the Configuration area from the **SSO Configuration** list, select the SAML IdP service that is bound to the SAML SP connector with the resources you want.
6. In the **Customization Settings for English** area in the **Caption** field, type a caption for this SAML resource.
7. Click **Finished**.
   The SAML resource is created and associated with a SAML IdP service that is bound to one external service provider.

Configuring a full webtop

A full webtop allows your users to connect and disconnect from a network access connection, portal access resources, SAML resources, app tunnels, remote desktops, and administrator-defined links.

1. On the Main tab, click **Access** > **Webtops** > **Webtop Lists**.
   The Webtops screen displays.
2. Click **Create**.
   The New Webtop screen opens.
3. In the **Name** field, type a name for the webtop.
4. From the **Type** list, select **Full**.
   The Configuration area displays with additional settings configured at default values.
5. Click **Finished**.

The webtop is now configured, and appears in the list. You can edit the webtop further, or assign it to an access policy.

To use this webtop, it must be assigned to an access policy with an advanced resource assign action or with a webtop, links, and sections assign action. All resources assigned to the full webtop are displayed on the full webtop.

Configuring an access policy for a SAML SSO portal

Before you configure this access policy, configure an access profile without selecting an SSO configuration for it.

Configure an access policy so that the BIG-IP® system, as a SAML Identity Provider (IdP) can authenticate users using any non-SAML authentication type, and assign SAML resources and a webtop to the session.

***Note:** This access policy supports users that initiate a connection at a SAML service provider or at the SAML IdP.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Add one or more authentication checks on the fallback branch after the **Logon Page** action.

   Select the authentication checks that are appropriate for application access at your site.
7. On a successful branch after an authentication check, assign SAML resources and a full webtop to the session.
   a) Click plus **[+]** on a successful branch after an authentication check.
      The Add Item window opens.
   b) On the Assignment tab, select the **Advanced Resource Assign** agent, and click **Add Item**.
      The Resource Assignment window opens.
   c) Click **Add new entry**.
      An **Empty** entry displays.
   d) Click the **Add/Delete** link below the entry.
      The screen changes to display resources on multiple tabs.
   e) Select the SAML tab, then from it select the SAML resources that represent the service providers that authorized users can access.
   f) Click **Update**.
      The window changes to display the Properties screen, where the selected SAML resources are displayed.
   g) Click the **Add/Delete** link below the entry.
      The screen changes to display resources on multiple tabs.
   h) Select the Webtop tab, then select a full webtop on which to present the selected resources.

      You must assign a full webtop to the session even if you have configured all SAML resources to not publish on a webtop.
   i) Click **Update**.

The window changes to display the Properties screen. The selected webtop and SAML resources are displayed.

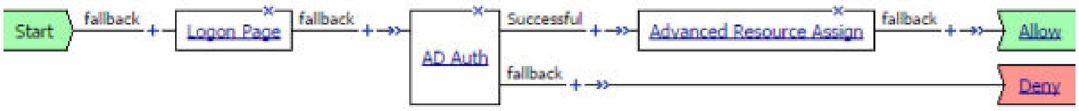
j) Click **Save**.
The Properties window closes and the Access Policy window is displayed.

You have configured a webtop to display resources that are available from service providers and that an authorized user can access.

8. (Optional) Add any other branches and actions that you need to complete the policy.
9. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
10. Click the **Apply Access Policy** link to apply and activate the changes to the policy.
11. Click the **Close** button to close the visual policy editor.

You have an access policy that presents a logon page, authenticates the user, and assigns SAML resources and a full webtop on which to present them to the user.

Simple access policy for access to services on SAML service providers



To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Adding IdP metadata from APM to external SAML SPs

To complete the agreement between Access Policy Manager® as the SAML IdP and a SAML Service Provider (SP), you must configure IdP metadata at the service provider.

Note: *Complete this step on each SAML service provider for which an SP connector is bound to the SAML IdP service in APM®.*

Using the method that the vendor provides, either:

- Import the SAML IdP metadata file that you exported from APM for the SAML IdP service that this service provider uses.
- Or take information from the SAML IdP metadata file that you exported from APM for the SAML IdP service and add it to the service provider using the vendor's interface. Pay particular attention to the values for entityID, AssertionConsumerService, SingleSignOnService, and the certificate.

  Note: *Regardless of the value of entityID in the metadata file, type an SSO URI that consists of the virtual server host and /saml/idp/profile/redirectorpost/sso. For example, if the host virtual server is https://Bigip-idp, type:* `https://Bigip-idp/saml/idp/profile/redirectorpost/sso`

Using APM as a SAML IdP (no SSO portal)

Overview: Configuring a BIG-IP system as IdP for SP-initiated connections only

***Note:** A configuration that allows users to initiate connection from service providers (SPs) only, works only when all service providers require the same assertion type, and value, and the same attributes from the IdP.*

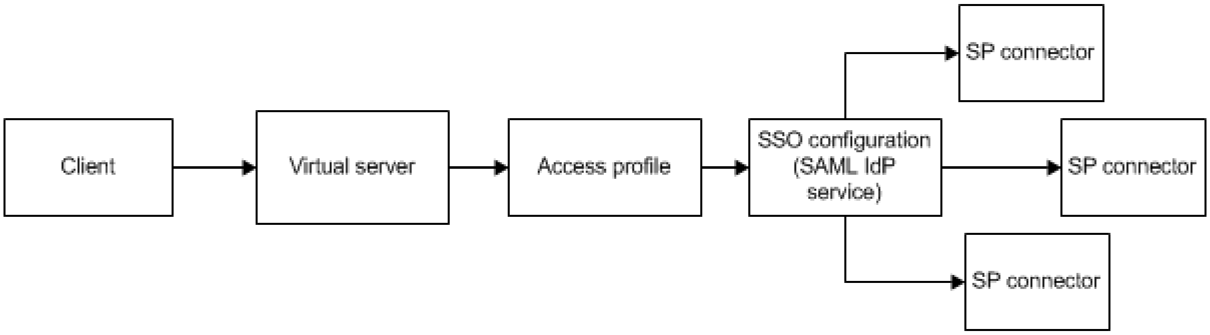
Configuration requirements for supporting SP-initiated connections only

For Access Policy Manager® as a SAML identity provider (IdP) to support only connections that start at a service provider, you need to meet these configuration requirements:

- SAML IdP services: One.
- SAML SP connectors: One for each SAML service provider.
- SSL certificate and key: One set for each SAML service provider, imported into the store on the BIG-IP® system.
- An access profile.
- An access policy.
- A virtual server that assigns the access profile.

Configuration requirements are summarized in this diagram.

Figure 25: Configuration requirements for supporting SP-initiated connections



About local IdP service

A *SAML IdP service* is a type of single sign-on (SSO) authentication service in Access Policy Manager® (APM®). When you use a BIG-IP® system as a SAML identity provider (IdP), a SAML IdP service provides SSO authentication for external SAML service providers (SPs). You must bind a SAML IdP service to SAML SP connectors, each of which specifies an external SP. APM responds to authentication requests from the service providers and produces assertions for them.

About SP connectors

A SAML service provider connector (an SP connector) specifies how a BIG-IP® system, configured as a SAML Identity Provider (IdP), connects with an external service provider.

What are the available ways I can configure a SAML SP connector?

You can use one or more of these methods to configure SAML service provider (SP) connectors in Access Policy Manager®.

- From metadata - Obtain a metadata file from the vendor and import it into Access Policy Manager. The advantage to this method is that the vendor provides the majority of all required data, including certificates. You can complete the configuration by simply typing a unique name for the SP connector, a very few additional required fields, and browsing to and importing the file. Access Policy Manager then configures the SP connector.
- From template - Use templates that Access Policy Manager provides for some vendors; for example, Google. The advantages to this method are that:

  - Most required data is included in the template
  - Additional required data is minimal. You can obtain it and certificates from the vendor

  After you select a template and type data into a few fields, Access Policy Manager configures the SP connector.
- Custom - Obtain information from the vendor and type the settings into the Configuration utility. To use this method, you must also obtain certificates from the vendor and import them into the BIG-IP® system. Use this method when a metadata file or a template for an SP connector is not available.

Task summary

Setting up a BIG-IP® system as a SAML identity provider (IdP) system involves two major activities:

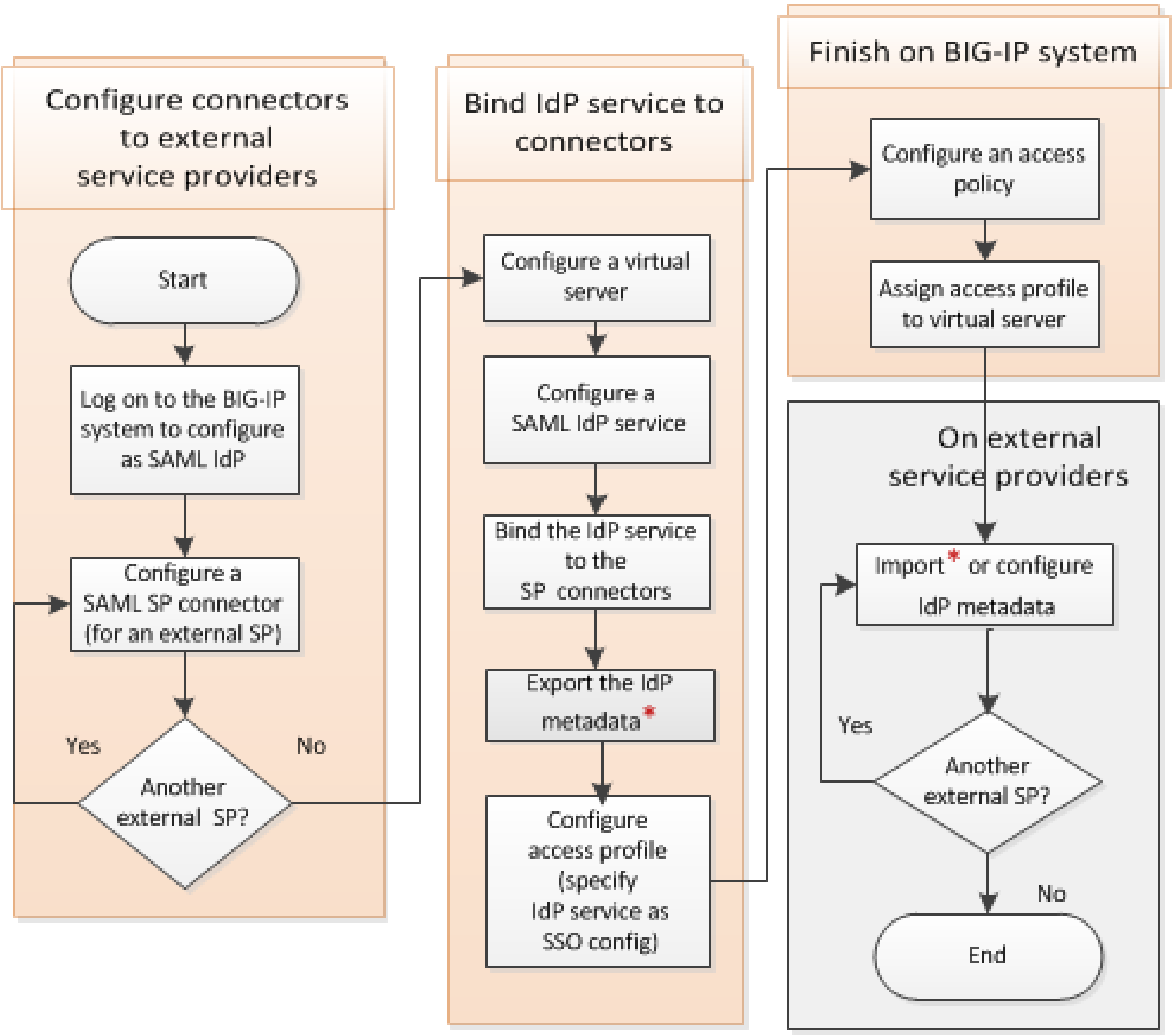
- First, you set up connection from the BIG-IP system to the external SAML service providers (SPs)
- Then, you set up connection from the external SAML SPs to the BIG-IP system

Task list

Creating a virtual server for a BIG-IP (as SAML IdP) system
Configuring an artifact resolution service
Configuring SAML SP connectors
Configuring a SAML IdP service
Binding a SAML IdP service to multiple SP connectors
Exporting SAML IdP metadata from APM
Creating an access profile associated with the SAML IdP service
Verifying log settings for the access profile
Configuring an access policy to provide authentication from the local IdP
Adding the access profile to the virtual server
Adding IdP metadata from APM to external SAML SPs

Flowchart: Configuration to support SP-initiated connections only

This flowchart illustrates the process for configuring a BIG-IP® system as a SAML identity provider (IdP) without providing an SSO portal.


Configure connectors to external service providers
Start
Log on to the BIG-IP system to configure as SAML IdP
Configure a SAML SP connector (for an external SP)
Yes
No
Another external SP?
Bind IdP service to connectors
Configure a virtual server
Configure a SAML IdP service
Bind the IdP service to the SP connectors
Export the IdP metadata*
Configure access profile (specify IdP service as SSO config)
Finish on BIG-IP system
Configure an access policy
Assign access profile to virtual server
On external service providers
Import* or configure IdP metadata
Yes
Another external SP?
No
End

Creating a virtual server for a BIG-IP (as SAML IdP) system

Before you start this task, configure a client SSL profile and a server SSL profile if you are going to create an SSL virtual server.

***Note:** Access Policy Manager® supports using a non-SSL virtual server for the BIG-IP® system configured as a SAML Identity Provider (IdP). However, we recommend using an SSL virtual server for security reasons. The following procedures include steps that are required for configuring an SSL virtual server, such as selecting client and server SSL profiles, and setting the service port to HTTPS.*

Specify a host virtual server to use as the SAML IdP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.

5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an IdP now appears on the Virtual Server List. The virtual server destination is available for use in one or more SAML IdP service configurations.

Configuring an artifact resolution service

Before you configure the artifact resolution service (ARS), you need to have configured a virtual server. That virtual server can be the same as the one used for the SAML Identity Provider (IdP), or you can create an additional virtual server.

Note: *F5® highly recommends that the virtual server definition include a server SSL profile.*

You configure an ARS so that a BIG-IP® system that is configured as a SAML IdP can provide SAML artifacts in place of assertions. With ARS, the BIG-IP system can receive Artifact Resolve Requests (ARRQ) from service providers, and provide Artifact Resolve Responses (ARRP) for them.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **Artifact Resolution Services**.
2. Click **Create**.
   The Create New SAML Artifact Resolution Service popup screen opens, showing general settings.
3. In the **Name** field, type a name for the artifact resolution service.
4. In the **Description** field, type a new description.
5. Click **Service Settings.**
6. From the **Virtual Server** list, select the virtual server that you created previously.

   ARS listens on the IP address and port configured on the virtual server.
7. In the **Artifact Validity (Seconds)** field, type the number of seconds for which the artifact remains valid. The default is 60 seconds.

   The BIG-IP® system deletes the artifact if the number of seconds exceeds the artifact validity number.
8. For the **Send Method** setting, select the binding to use to send the artifact, either **POST** or **Redirect**.
9. In the **Host** field, type the host name defined for the virtual server, for example **ars.siterequest.com**.
10. In the **Port** field, type the port number defined in the virtual server. The default is `443`.
11. Click **Security Settings.**
12. To require that artifact resolution messages from an SP be signed, select the **Sign Artifact Resolution Request** check box.
13. To use HTTP Basic authentication for artifact resolution request messages, in the **User Name** field, type a name for the artifact resolution service request and in the **Password** field, type a password.

    These credentials must be present in all Artifact Resolve Requests sent to this ARS.
14. Click **OK**.
    The popup screen closes, leaving the Artifact Resolution Services list screen open.

The Artifact Resolution Service is ready for use.

Configuring SAML SP connectors

Before you can configure a SAML service provider, you must first obtain an SSL certificate from the SAML service provider (SP) and import it into the certificate store on the BIG-IP® system.

You configure information about a SAML service provider so that Access Policy Manager® (APM®) can act as a SAML Identity Provider (IdP) for it.

***Note:** Configure one SAML SP connector for each external SAML service provider for which this BIG-IP system provides SSO authentication service.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **External SP Connectors**.
   A list of SAML SP connectors displays.
2. Click **Create**.
   The Create New SAML SP Connector screen opens.
3. In the **Service Provider Name** field, type a unique name for the SAML SP connector.
4. In the **SP Entity ID** field, type a unique identifier for the service provider.

   This is usually a unique URI that represents the service provider. You should obtain this value from the service provider.
5. From the left pane, select **Endpoint Settings**.

   The appropriate settings are displayed.
6. (Optional) In the **Relay State** field, type a value.

   The relay state can be an absolute path, such as `/hr/index.html`; it can be a URL, such as `https://www.abc.com/index.html`; or, it can be anything that the service provider understands. The information passed in relay state could be used by the service provider according to business logic. For example, some service providers use relay state to maintain a session state, while others use it to perform an action, such as redirecting the user to the page passed in relay state. APM sends the relay state value back to the service provider as part of the assertion response in the `RelayState` parameter.

   When the `RelayState` parameter is already part of the authentication request to the BIG-IP system, APM returns the value that was sent in the request. Otherwise, APM uses the value from this configuration.
7. In the Assertion Consumer Services area, specify at least one assertion consumer service.

   A service provider can use multiple bindings to receive an assertion from the Identity Provider. The service provider can specify a different assertion consumer service (ACS) URL for each binding, and provide a unique ACS URL index for the binding.

   To support SAML artifacts, make sure that at least one ACS specifies the artifact binding.

   a) Click **Add**.
      A new row displays in the table.
   b) In the **Index** field, type the index number, zero (0) or greater.
   c) If this is the default service, select the **Default** check box.

      You must specify one of the services as the default.
   d) In the **Location URL** field, type the URL where the IdP can send an assertion to this service provider.

      APM supports HTTP-Artifact binding, POAS (HTTP reverse SOAP) binding, and HTTP-POST binding to this service.
   e) From the **Binding** list, select **Artifact**, **PAOS**, or **POST**.
   f) Click **Update**.

8. From the left pane, select **Security Settings**.
   a) If the SP should sign the authentication or the artifact resolution requests that it sends to the SAML IdP (this BIG-IP system), select the **Require Signed Authentication Request** check box, select a private key from the **Message Signing Private Key** list, and select a certificate from the **Message Signing Certificate** list.

      This device (BIG-IP system as IdP) uses the certificate to verify the signature of the request from the SP.
   b) To require that the SAML IdP sign the assertion before sending it to the SP, select the **Assertion must be signed** check box, and select an algorithm from the **Signing Algorithm** list.

      Assertion must be signed is selected by default. Clearing this check box is not recommended.
   c) To require that the SAML IdP sign the response before sending it to the SP, select the **Response must be signed** check box.

      Note: *The algorithm specified in the* ***Signing Algorithm*** *list applies to a signed assertion and a signed response.*
   d) To require that the SAML IdP encrypt the assertion before sending it to the SP, select the **Assertion must be encrypted** check box, select a type from the **Encryption Type** list, and select a certificate from the **Encryption Certificate** list.

      APM supports AES128, AES192, and AES256 encryption types.
9. From the left pane, select **SLO Service Settings**.

   SLO stands for Single Logout.
   a) (Optional) In the **Single Logout Request URL** field, type a URL specifying where APM should send a logout request to this service provider when the BIG-IP system initiates a logout request.
   b) In the **Single Logout Response URL** field, type a URL to which the SP should send a logout response for the BIG-IP system to indicate that single logout is complete.

   Note: *APM supports HTTP-POST binding for the SLO service. For SLO to work, all entities (SPs and IdPs), must support SLO.*
10. Click **OK**.
    The popup screen closes.

APM creates a SAML SP connector. It is available to bind to a SAML IdP service.

Configuring a SAML IdP service

Configure a SAML Identity Provider (IdP) service for the BIG-IP® system, configured as a SAML IdP, to provide authentication service for SAML service providers (SPs).

Note: *Configure this IdP service to meet the requirements of all SAML service providers that you bind with it.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Click **Create**.
   The Create New IdP Service popup screen displays.
3. In the **IdP Service Name** field, type a unique name for the SAML IdP service.
4. In the **IdP Entity ID** field, type a unique identifier for the IdP (this BIG-IP® system).

   Typically, the ID is a URI that points to the BIG-IP virtual server that is going to act as a SAML IdP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://siterequest.com/idp`, where the path points to the virtual server you use for BIG-IP system as a SAML IdP.

5. If the **IdP Entity ID** field does not contain a valid URI, you must provide one in the IdP Name Settings area:
   a) From the **Scheme** list select **https** or **http**.
   b) In the **Host** field, type a host name.
   For example, type `siterequest.com` in the **Host** field.
6. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
7.
8. If you select **SAML Profiles** on the left pane, the **Web Browser SSO** check box is selected by default.

   At least one profile must be selected.
9. To specify that this IdP use an artifact resolution service, click **Endpoint Settings** on the left pane and select a service from the **Artifact Resolution Service** list.
10. On the left pane, select **Assertion Settings** and complete the settings that display:
    a) From the **Assertion Subject Type** list, select the type of subject for the IdP to authenticate.
    b) From the **Assertion Subject Value** list, select the name of a session variable.

    This variable, *`%{session.logon.last.username}`*, is generally applicable. Some session variables are applicable depending on the type of authentication that you use for your site.
    c) In the **Authentication Context Class Reference** field, select a URI reference.

    The URI reference identifies an authentication context class that describes an authentication context declaration.
    d) In the **Assertion Validity (in seconds)** field, type the number of seconds for which the assertion is valid.
    e) To encrypt the subject, select the **Enable encryption of Subject** check box.
    The **Encryption Strength** list becomes available.
    f) From the **Encryption Strength** list, select a value.

    Supported values are AES128, AES192, and AES256.
11. On the left pane, select **SAML Attributes**, and for each attribute that you want to include in the attribute statement, repeat these substeps.
    a) Click **Add**.
    A Create New SAML Attribute popup screen displays.
    b) In the **Name** field, type a unique name for the attribute.

    Usually, the name is a fixed string, but it can be a session variable.
    c) To add a value to the attribute, click **Add**, type a value in the **Value(s)** field, and click **Update** to complete the addition.

    You can use a session variable for the value.

    This example shows using a fixed string for the name and a session variable for the value. Name: `user_telephonenumber` and value: `%{session.ad.last.attr.telephoneNumber}`.

    You can repeat this step to add multiple values for an attribute.
    d) To encrypt the values, select the **Encrypt** check box and select a value from the **Type** list.

    Supported values for type are AES128, AES192, and AES256.
    e) Click **OK**.
    The Create New SAML Attribute popup screen closes.
12. Click **Security Settings** from the left pane.
    a) From the **Signing Key** list, select the key from the BIG-IP system store.

    None is selected by default.

b) From the **Signing Certificate** list, select the certificate from the BIG-IP system store.

When selected, the IdP (the BIG-IP system) publishes this certificate to the service provider so the service provider can verify the assertion. **None** is selected by default.

13. Click **OK**.
The popup screen closes. The new IdP service appears on the list.

Access Policy Manager® (APM®) creates a SAML IdP service. It is available to bind to SAML SP connectors. This service works with external service providers that share the same requirements for assertion settings and SAML attribute settings.

Binding a SAML IdP service to multiple SP connectors

Select a SAML Identity Provider (IdP) service and the SAML service provider (SP) connectors that use the service so that this BIG-IP® system can provide authentication (SAML IdP service) to external SAML service providers.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
The Local IdP Services screen opens.
2. Select a SAML IdP service from the list.

   A SAML IdP service provides authentication service.
3. Click **Bind/Unbind SP Connectors**.
The screen displays a list of available SAML SP connectors.
4. Select only the SAML SP connectors that you want to use this service.
5. Click **OK**.
The screen closes.

The SAML IdP service is bound to the SAML service providers specified in the SAML SP connectors.

Exporting SAML IdP metadata from APM

You need to convey the SAML Identity Provider (IdP) metadata from Access Policy Manager® (APM®) to the external service providers that use the SAML IdP service. Exporting the IdP metadata for a SAML IdP service to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
The Local IdP Services screen opens.
2. Select a SAML IdP service from the table and click **Export Metadata**.
A popup screen opens, with **No** selected on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

      APM uses the key to sign the metadata.
   c) From the **Signature Verification Certificate** list, select a certificate.

      APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.
4. Select **OK**.
APM downloads an XML file.

An XML file that contains IdP metadata is available.

Creating an access profile associated with the SAML IdP service

Use this procedure when this BIG-IP® system, as a SAML Identity Provider (IdP), supports service provider-initiated connections only.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. In the SSO Across Authentication Domains (Single Domain mode) area, from the **SSO Configuration** list, select the name of the local SAML IdP service.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy to provide authentication from the local IdP

Configure an access policy so that this BIG-IP® system, as a SAML Identity Provider (IdP) can provide authentication for SAML service providers.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.

3. Click the **(+)** icon anywhere in the access policy to add a new item.

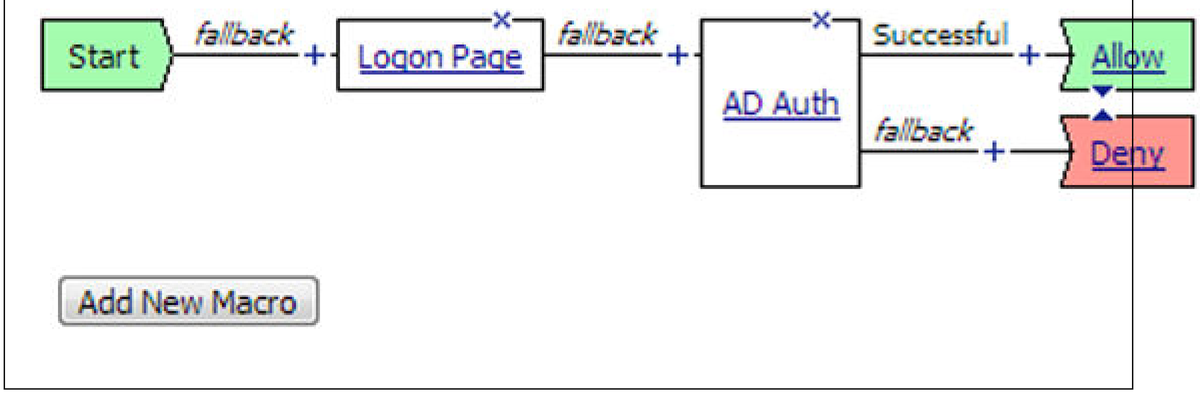
   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Add one or more authentication checks on the fallback branch after the **Logon Page** action.

   Select the authentication checks that are appropriate for application access at your site.
7. (Optional) Add any other branches and actions that you need to complete the policy.
8. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
9. Click the **Apply Access Policy** link to apply and activate the changes to the policy.
10. Click the **Close** button to close the visual policy editor.

You have an access policy that presents a logon page and authenticates the user..

Access policy to provide authentication for SAML service providers when this BIG-IP system is the IdP



To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Adding IdP metadata from APM to external SAML SPs

To complete the agreement between Access Policy Manager® as the SAML IdP and a SAML Service Provider (SP), you must configure IdP metadata at the service provider.

***Note:** Complete this step on each SAML service provider for which an SP connector is bound to the SAML IdP service in APM®.*

Using the method that the vendor provides, either:

- Import the SAML IdP metadata file that you exported from APM for the SAML IdP service that this service provider uses.
- Or take information from the SAML IdP metadata file that you exported from APM for the SAML IdP service and add it to the service provider using the vendor's interface. Pay particular attention to the values for entityID, AssertionConsumerService, SingleSignOnService, and the certificate.

  ***Note:** Regardless of the value of entityID in the metadata file, type an SSO URI that consists of the virtual server host and /saml/idp/profile/redirectorpost/sso. For example, if the host virtual server is https://Bigip-idp, type:* `https://Bigip-idp/saml/idp/profile/redirectorpost/sso`

Using APM as a SAML Service Provider

About configuration requirements for APM as a SAML service provider

For Access Policy Manager® to act as a SAML service provider (SP), you must create this configuration.

- SAML SP service - One.
- SAML Identity Provider (IdP) connectors - One or more.
- An SSL certificate and key from each SAML IdP, imported into the store on the BIG-IP® system.
- An access profile.
- An access policy that includes the SAML Auth agent.
- A virtual server that assigns the access profile.

About local SP service

A *SAML SP service* is a type of AAA service in Access Policy Manager® (APM®). It requests authentication from an external SAML Identity Provider (IdP) that is specified on APM in a SAML IdP connector. (You bind a SAML service provider (SP) service to one or more SAML IdP connectors.) APM requests authentication from an IdP and consumes assertions from it to allow access to resources behind APM.

About SAML IdP discovery

On a BIG-IP® system that you use as a SAML service provider (SP), you can bind an SP service to one or more SAML Identity Provider (IdP) connectors (each of which specifies an external IdP). When you bind an SP service to multiple IdP connectors, Access Policy Manager® chooses the correct IdP connector at run time through a filtering and matching process called IdP discovery.

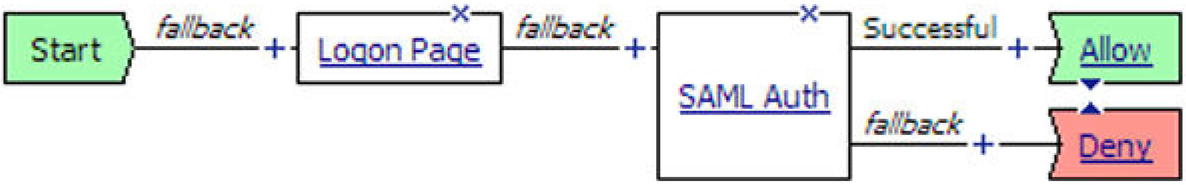
Scenario

You might bind multiple IdP connectors to an SP service on the BIG-IP system when you must provide services to different businesses and universities, each of which specifies an IdP to identify their users. When the user's information arrives at the SP service on the BIG-IP system, the SP service identifies the correct IdP and redirects the user to authenticate against that IdP before the SP service provides access to the service.

Note: *The SP service performs IdP discovery for a user only when the user initiates connection from an SP.*

Session variables and the typical access policy for BIG-IP system as SP

On a BIG-IP system configured as an SP, the typical access policy presents a logon page to the user. The Logon Page action populates session variables. You can customize the Logon Page action and affect session variable values. A SAML Auth action follows the logon page.



A SAML Auth action specifies an SP service. An SP service is an AAA service that requests authentication from an external IdP (specified in an IdP connector).

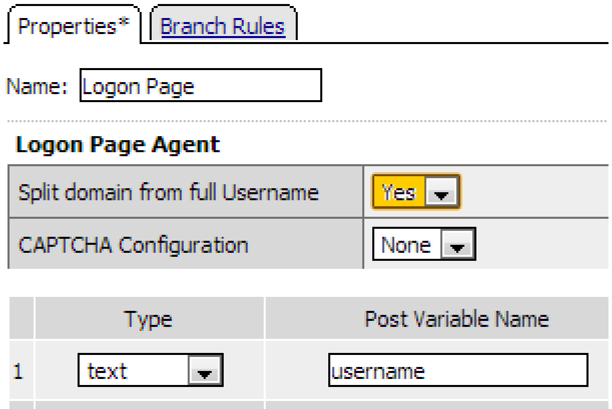
Session variables and SAML IdP discovery

Among multiple IdP connectors, the BIG-IP system must discover the correct external IdP with which to authenticate a user. For IdP discovery to work, you must specify matching criteria, a session variable name and value, for each IdP connector.

For example, users of a service might go to a particular landing page. When you bind the IdP connector, for the external IdP that serves those users, to the SP service, select the `%{session.server.landinguri}` session variable and supply a landing path value, such as, `/south*`. For users going to URLs such as `https://sp-service/southwest` and `https://sp-service/southeast`, the SP service selects the same IdP to authenticate them.

Logon Page action customization

These are some common customization examples for the Logon Page action.



Name: Logon Page

Logon Page Agent

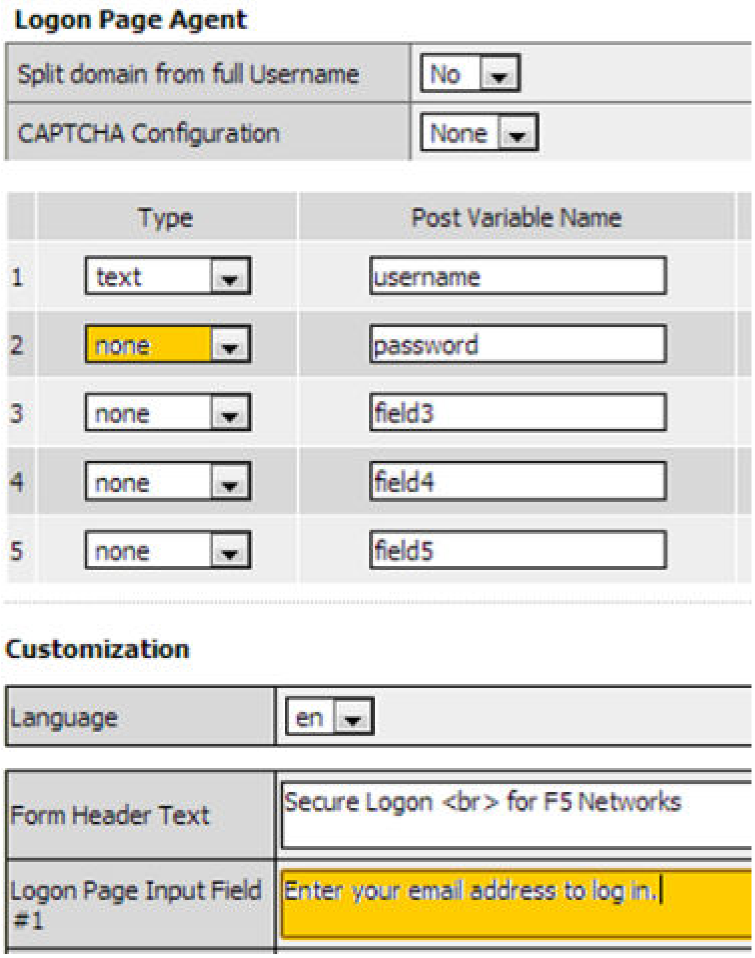
Split domain from full Username	Yes
CAPTCHA Configuration	None

	Type	Post Variable Name
1	text	username

Figure 26: Setting the value of session.logon.last.domain variable to the domain name only

Select **Yes** for **Split domain from full Username**. The Logon Page agent takes the user name, such as joe@office.com, that was entered and creates the following session variables with these values.

Session Variable	Value
%{session.logon.last.username}	joe
%{session.logon.last.domain}	office.com
%{session.logon.last.logonname}	joe@office.com



Logon Page Agent

Split domain from full Username	No
CAPTCHA Configuration	None

	Type	Post Variable Name
1	text	username
2	none	password
3	none	field3
4	none	field4
5	none	field5

Customization

Language	en

Form Header Text	Secure Logon for F5 Networks
Logon Page Input Field #1	Enter your email address to log in.

Figure 27: Obtaining and email address as the username

Change the prompt for the first text field (username field). To omit the password: for **Type**, select **none** from the list.

About IdP connectors

An IdP connector specifies how a BIG-IP® system, configured as a SAML service provider (SP), connects with an external SAML identity provider (IdP).

About methods for configuring SAML IdP connectors in APM

You can use one or more of these methods to configure SAML identity provider (IdP) connectors in Access Policy Manager® (APM®).

- From metadata - Obtain a metadata file from the vendor and import it into APM. The advantage to this method is that the vendor provides all required data, including the certificate. You can complete the configuration by simply typing a unique name for the identity provider, and browsing to and importing the file. APM imports the certificate to the BIG-IP® system and configures the SAML IdP connector.
- From template - Use templates that APM provides for some vendors. The advantages to this method are that:
  - Most required data is included in the template. (Note that the certificate is not included.)
  - Additional required data is minimal and is available from the vendor.

APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- Custom - Research the identity provider requirements and type all settings into the Configuration utility. Use this method when a metadata file or a template for an identity provider is not available. APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- IdP Automation - Provide files with cumulative IdP metadata on remote systems, then configure BIG-IP IdP automation to poll the files periodically and create IdP connectors and bind them to a specific service provider (SP) service.

Task summary

Setting up a BIG-IP® system as a SAML service provider (SP) involves two activities:

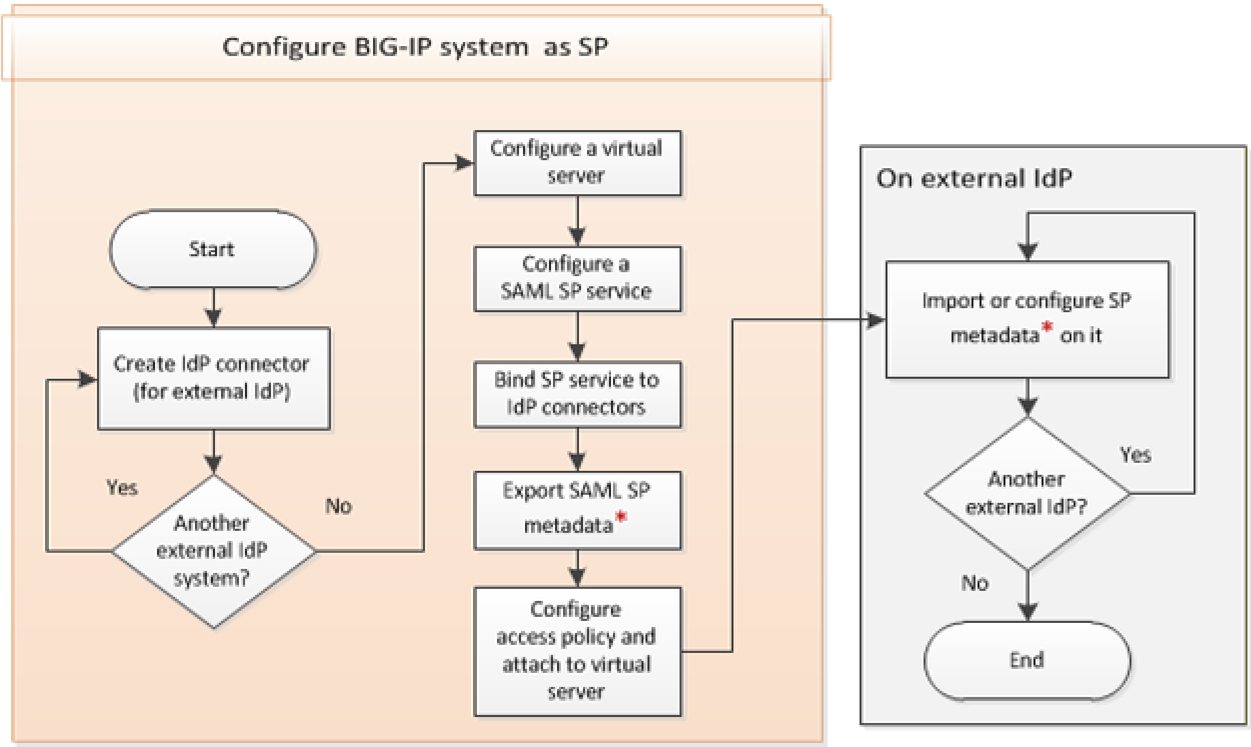
- First, you set up one BIG-IP system as a SAML service provider (SP) system
- Then, you go to one or more external SAML identity provider (IdP) systems and set up connectivity to the SP system

Task list

Configuring a custom SAML IdP connector
Creating a virtual server for a BIG-IP (as SAML SP) system
Configuring a SAML SP service
Binding a SAML SP service to SAML IdP connectors
Exporting SAML SP metadata from APM
Configuring an access policy to authenticate with an external SAML IdP
Verifying log settings for the access profile
Adding the access profile to the virtual server
Adding SAML SP metadata from APM to an external SAML IdP
Creating SAML authentication context classes

Flowchart: BIG-IP system as a SAML service provider configuration

This flowchart illustrates the process for configuring a BIG-IP® system as a SAML service provider (SP). In this configuration, the BIG-IP system relies on external SAML Identity Providers (IdPs).


Configure BIG-IP system as SP
Start
Create IdP connector (for external IdP)
Another external IdP system?
Yes
No
Configure a virtual server
Configure a SAML SP service
Bind SP service to IdP connectors
Export SAML SP metadata*
Configure access policy and attach to virtual server
On external IdP
Import or configure SP metadata* on it
Another external IdP?
Yes
No
End

Configuring a custom SAML IdP connector

You configure a SAML IdP connector so that Access Policy Manager® (APM®) (as a SAML service provider) can send authentication requests to this Identity Provider (IdP), relying on it to authenticate users and to provide access to resources behind APM.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider** > **External IdP Connectors**.
   The External IdP Connectors screen displays.
2. Click **Create** > **Custom**.
   The Create New SAML IdP Connector screen opens.
3. In the **Name** field, type a unique name for the SAML IdP connector.
4. In the **IdP Entity ID** field, type a unique identifier for the IdP.

   This is usually a URI. Obtain this value from the vendor.
5. To configure single sign-on service, from the left pane, select **Endpoint Settings** > **Single Sign On Service Settings**.
   The screen changes to display the applicable settings.
6. In the **Single Sign On Service URL** field, type the location on the IdP where APM should send authentication requests.
7. (Optional) From the **Single Sign On Service Binding** field, select one:

   - POST (the default value)
   - Redirect

   This is the binding APM uses to send authentication requests to the IdP.
8. For the service provider to connect to an artifact resolution service and exchange an artifact for an assertion, select **Endpoint Settings** > **Artifact Resolution**.
   a) In the Artifact Resolution Service Settings area, in the **Location URL** field, type the URI of the IdP artifact resolution service.

The URI must include the scheme, host name, port, and full path.

b) In the **IP Address** field, type the IP address that this BIG-IP® system (as SP) will use to connect to the IdP artifact resolution service.

The value must be a valid IPv4 or IPv6 address.

Note: *The host name from the* ***Location URL*** *must resolve to this IP address.*

---

c) In the **Port** field, type the port for the artifact resolution service.

This must match the port number from the **Location URL**.

d) To specify that the IdP requires that artifact resolve requests be signed, select the **Sign Artifact Resolution Request** check box, and select a profile from the **Server SSL Profile** list.

e) If the artifact resolution service is protected by HTTP Basic authentication, in the **User Name** field, type a Basic user name and in the **Password** field type a password.

9. Select **Assertion Settings** from the left pane.

10. From the **Identity Location** list, select where to find the *principal* (usually, this is a user) to be authenticated:

- **Subject** - In the subject of the assertion. This is the default setting.
- **Attribute** - In an attribute. If selected, the **Identity Location Attribute** field displays, and you must type an attribute name into it.

Note: *If the assertion from the IdP does not include this attribute, the BIG-IP system (as SP) does not accept the assertion as valid.*

---

11. Select **Security Settings** from the left pane.

a) (Optional) To require that the SAML SP sign the assertion request before sending it to the IdP, select the **Must be signed** check box and select an algorithm from the **Signing Algorithm** list.

b) From the Certificate Settings area, select a certificate from the **IdP's Assertion Verification Certificate** list.

The BIG-IP system uses this certificate from the IdP to verify the signature of the assertion from the IdP. If the certificate from the IdP is not in the BIG-IP system store, obtain it and import it into the store. Then edit this IdP connector to select the certificate for it.

12. Select **SLO Service Settings** from the left pane.

a) (Optional) In the **Single Logout Request URL** field, type a URL.

When a service provider initiates a logout, APM sends the logout request to the SAML Identity Provider (IdP) using this URL.

b) In the **Single Logout Response URL** field, type a URL.

When the IdP initiates a logout, APM sends the logout response to the IdP using this URL.

Note: *APM supports HTTP-POST binding for the SLO service. For SLO to work, all entities (SPs and IdPs) must support SLO.*

---

13. Click **OK**.
The popup screen closes.

APM creates a SAML IdP connector. It is available to bind to a SAML SP service.

Creating a virtual server for a BIG-IP (as SAML SP) system

Specify a host virtual server to use as the SAML SP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
The Virtual Server List screen opens.

2. Click the **Create** button.

The New Virtual Server screen opens.

3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an SP now appears on the Virtual Server List. The virtual server destination is available for use in a SAML SP service configuration.

Configuring a SAML SP service

Configure a SAML service provider (SP) service for Access Policy Manager® to provide AAA authentication, requesting authentication and receiving assertions from a SAML identity provider (IdP).

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Click **Create**.
   The Create New SAML SP Service screen opens.
3. In the **Name** field, type a unique name for the SAML SP service.
4. In the **Entity ID** field, type a unique identifier for the service provider.

   Typically entity ID is a URI that points to the BIG-IP virtual server that is going to act as SAML SP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://bigip-sp`, where `https:/bigip-sp` points to the virtual server you use for BIG-IP system as a SAML service provider.
5. If the **Entity ID** field does not contain a valid URI, in the SP Name Settings area from the **Scheme** list, select **https** or **http** and in the **Host** field, type a host name.
   For example, type `siterequest.com` in the **Host** field.
6. In the **Relay State** field, type a value.

   The value can be an absolute path, such as `hr/index.html` or a URI, such as `https://www.abc.com/index.html`. It is where the service provider redirects users after SAML single sign-on completes.
7. For this service provider to request an artifact instead of an assertion from the IdP, from the left pane select **Endpoint Settings** and, from the **Assertion Consumer Service Binding** list, select **Artifact**.

   POST is the default setting.
8. From the left pane, select **Security Settings**.

   The screen displays the applicable settings.
9. If you want this BIG-IP system to send signed authentication requests to the SAML IdP, select **Signed Authentication Request**. Then select a key and a certificate from those in the BIG-IP system store from the **Message Signing Private Key** and **Message Signing Certificate** lists.
10. If this BIG-IP system requires signed assertions from the SAML IdP, ensure that the **Want Signed Assertion** check box remains selected.

11. If this BIG-IP system requires encrypted assertions from the SAML IdP, select **Want Encrypted Assertion**. Then select a key and a certificate from those in the BIG-IP system store from the **Assertion Decryption Private Key** and **Assertion Decryption Certificate** lists.

    The BIG-IP system uses the private key and certificate to decrypt the assertion.
12. To configure additional service provider attributes, from the left pane click **Advanced**.

    The screen displays the applicable settings.
13. To force users to authenticate again even when they have an SSO session at the identity provider, select the **Force Authentication** check box.

    This setting is for use when the external IdP supports a force authentication flag.
14. To allow the external IdP, when processing requests from this BIG-IP system as SP, to create a new identifier to represent the principal, select the **Allow Name-Identifier Creation** check box.
15. To specify the type of identifier information to use, select a URI reference from the **Name-Identifier Policy Format** list.
    For example, if a Service Provider (SP) initiates SSO by sending an `AuthnRequest` to the IdP with format **urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress**, then the IdP response should contain the subject identity in email format.
16. To specify that the assertion subject's identifier be returned in the namespace of an SP other than the requester, or in the namespace of a SAML affiliation group of SPs, type a value in the **SP Name-Identifier Qualifier** field.
17. Click **OK**.
    The screen closes.

APM® creates the SAML SP service. It is available to bind to SAML IdP connectors and to export to a metadata file.

Binding a SAML SP service to SAML IdP connectors

Select a SAML SP service and bind one or more SAML IdP connectors to it so that this device (BIG-IP® system as a SAML service provider) can request authentication from the appropriate external IdP.

***Note:** If you bind this SP service to more than one IdP connector, you must configure matching criteria for each IdP connector. When users initiate connections at service providers, the BIG-IP system uses matching criteria to identity the correct IdP among many using SAML IdP discovery.*

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select a SAML SP service from the list.
3. Click **Bind/Unbind IdP Connectors**.
   A pop-up screen displays a list of any IdP connectors that are associated with this SP service.
4. To add an SAML IdP connector to the list, click **Add New Row**.
5. To bind only one IdP connector with this SP service, complete the configuration:
   a) Select a connector from the **SAML IdP Connectors** list in the new row.

      When you bind only one IdP connector to an SP service, you do not need to fill in the **Matching Source** and **Matching Value** fields.
   b) Click the **Update** button.

      The configuration is not saved until you click **OK**.
   c) Click **OK**.
      APM saves the configuration. The screen closes.
6. To bind multiple IdP connectors with this SP service, complete the configuration:
   a) Select a connector from the **SAML IdP Connectors** list in the new row.

b) In the **Matching Source** field, select or type the name of a session variable.

Use a session variable only if it is populated in the policy before the SAML Auth action.

For example, select **%{session.server.landinguri}** or type `%{session.logon.username}`.

c) In the **Matching Value** field, type a value.

The value can include the asterisk (*) wild card.

For example, type `*hibb*` or `south*` .

d) Click the **Update** button.

The configuration is not saved until you click **OK**.

e) To add other IdP connectors, start by clicking **Add New Row**, fill the new row, and end by clicking **Update**.

f) Click **OK**.
APM saves the configuration. The screen closes.

The SAML IdP connectors that you selected are bound the SAML SP service.

Exporting SAML SP metadata from APM

You need to convey the SP metadata from APM® to the external SAML IdP that provides authentication service to this SP. Exporting the SAML SP metadata to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select an SP service from the list and click **Export Metadata**.
   A popup window opens, displaying **No** on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

   APM uses the key to sign the metadata.

   c) From the **Signature Verification Certificate** list, select a certificate.

   APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.
4. Select **OK**.
   APM downloads an XML file.

You must either import the XML file on the IdP system or use the information in the XML file to configure SP metadata on the IdP system .

Configuring an access policy to authenticate with an external SAML IdP

Before you start this task, configure an access profile.

When you use this BIG-IP® system as a SAML service provider (SP), configure an access policy to direct users to an external SAML Identity Provider (IdP) for authentication.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

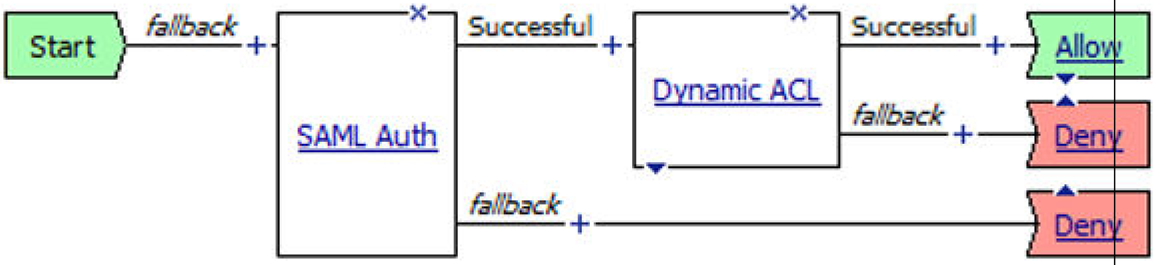
Note: Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.

A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.

4. On the Authentication tab, select **SAML Auth** and click the **Add Item** button.
   The SAML Auth properties window opens.
5. In the SAML Authentication SP area from the **AAA Server** list, select a SAML SP service and click **Save**.
   The Access Policy window displays.
6. Add any additional actions that you require to complete the policy.
7. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
8. At the top of the window, click the **Apply Access Policy** link to apply and activate your changes to this access policy.
9. Click the **Close** button to close the visual policy editor.

You have an access policy that uses SAML authentication against an external SAML IdP and further qualifies the resources that a user can access.

Simple access policy to authenticate users against an external SAML IdP



To apply this access policy to network traffic, add the access profile to a virtual server.

Note: To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: Log settings are configured in the ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.

4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Adding SAML SP metadata from APM to an external SAML IdP

To complete the agreement between APM® as the SAML service provider and a SAML IdP, you must configure service provider metadata at the IdP.

Note: *The method for configuring SAML service provider metadata at a SAML IdP will vary by vendor.*

Using the method that the vendor provides, either:

- Import the SAML SP metadata file that you exported from APM for a SAML SP service that is bound to the SAML IdP connector for this IdP.
- Or take information from the SAML SP metadata file that you exported from APM and add it using the vendor's interface. Pay particular attention to the values for entityID, AssertionConsumerService, and the certificate.

  Note: *Typically, the value of AssertionConsumerService is a URL that looks like this:* `https://bigip-sp-vs/saml/sp/profile/post/acs`.

Creating SAML authentication context classes

You create SAML authentication context classes to provide URIs to SAML service providers. These URIs specify authentication methods in SAML authentication requests and authentication statements.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Click **Authentication Context Classes**.
   The Authentication Context Classes screen displays.
3. Click **Create**.
   The Create New SAML Authentication Context Classes screen displays.
4. Click **General Settings**.
5. In the **Name** field, type a name used in the SAML Service Provider (SP) local SP service configuration.

6. In the **Description** field, type a descriptive text for the authentication context classes.
7. Click **Authentication Classes**.
8. Click **Add**.
9. In the **Name** field, type a name for the authentication class.
10. From the Value dropdown menu, select an existing value.
11. Add more values as needed.
12. Click **Update**.
13. Click **OK**.

An authentication context class with a list of authentication contexts is available.

Task summary
Adding SAML SP metadata from APM to an external SAML IdP
Task summary

Using BIG-IP® IdP Automation

Overview: Automating SAML IdP connector creation

When a BIG-IP® system is configured as a SAML service provider (SP), you can use SAML identity provider (IdP) automation to automatically create new SAML IdP connectors for SP services. Access Policy Manager® (APM®) polls a file or files that you supply; the files must contain cumulative IdP metadata. After polling, APM creates IdP connectors for any new IdPs and associates them with a specified SP service. APM uses matching criteria that you supply to send the user to the correct IdP.

When would I use SAML IdP automation?

Here is an example in which SAML Identity Provider (IdP) automation is especially useful. A large service provider (SP) supports a number of SAML identity providers. The service provider defines a SAML SP service on Access Policy Manager® (APM®) for access to that service. As IdPs come online, the service provider collects metadata from them and aggregates the IdP metadata into a file.

***Note:** The process for collecting and aggregating IdP metadata into a file is up to the service provider.*

APM polls the metadata file, creates IdP connectors, associates new connectors to the specified SAML SP service, and ensures that clients performing SP-initiated access are sent to the correct IdP.

Automating IdP connector creation for BIG-IP as SP

To create a BIG-IP® Identity Provider (IdP) automation configuration, you need a BIG-IP® system that is configured to function as a SAML service provider (SP) and you need to have SAML SP services defined.

You create a connector automation configuration to automatically create SAML IdP connectors and bind them to an SP service based on cumulative IdP metadata you maintain in a file or files. You specify matching criteria in connector automation for APM® to use, in order to send a user to the correct IdP.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider** > **Connector Automation**.
   The Connector Automation screen opens and displays a table. Each row includes a configuration name, the URLs where IdP metadata files are stored for a particular SP service, and the name of the SP service to which automation applies.
2. Click **Create**.
   The Create New SAML IdP Automation popup screen opens.
3. In the **Name** field, type a name for the IdP automation configuration.
4. For the **SP Service** setting, select a service from the list.

   If the SP service you want has not already been defined, click **Create** to configure it and add it to the list.

   APM periodically creates SAML IdP connectors and binds them to the SP service you specify here.
5. From the **IdP Matching Source** list, select or type the name of a session variable.

   At the time of SP-initiated SAML single sign-on, APM (as a SAML SP) matches the value of this session variable to the value in the tag that you specify in the **Metadata Tag Match Value** field.
6. In the **Metadata Tag Match Value** field, type the name of a metadata tag.

APM extracts the value in this tag from the IdP metadata and matches it with the value of the session variable specified in the **IdP Matching Source** field.

Note: Do not include any wildcard in the value.

7. In the **Metadata Tag For IdP Connector Name** field, type the name of a tag that is included in the IdP metadata.

   APM uses the value in the tag to name the IdP connector that it creates.
8. In the **Frequency** field, type a number of minutes.

   This specifies how often APM polls IdP metadata files.
9. Select **Metadata URLs** from the left pane.

   You specify URLs for one or more cumulative metadata files located on remote systems.

   A URL table displays in the right pane.
10. Specify a URL for each SAML IdP metadata file to be read. To add each URL, follow these steps:
    a) Click **Add**.
       A new field opens in the URL table.
    b) Type a URL.

       Begin the URL with `http` or `https`.

       For example, type `https://mywebsite.com/metdata/idp/idp_metadata.xml`.
    c) Click **Update**.
       The new URL displays in the top row of the table.
11. Click **OK**.

    The Create SAML IdP Automation screen closes. The new automation displays in the list.

For IdP automation to work, you must provide the metadata files as specified in the metadata URLs.

BIG-IP System Federation for SP-Initiated Connections

Overview: Federating BIG-IP systems for SAML SSO (without an SSO portal)

In a federation of BIG-IP® systems, one BIG-IP system acts as a SAML Identity Provider (IdP) and other BIG-IP systems act as SAML service providers (SPs).

This configuration supports:

- Only those connections that initiate at a service provider.
- Only service providers that accept assertions with similar subject type, attributes, and security settings.

About SAML IdP discovery

On a BIG-IP® system that you use as a SAML service provider (SP), you can bind an SP service to one or more SAML Identity Provider (IdP) connectors (each of which specifies an external IdP). When you bind an SP service to multiple IdP connectors, Access Policy Manager® chooses the correct IdP connector at run time through a filtering and matching process called IdP discovery.

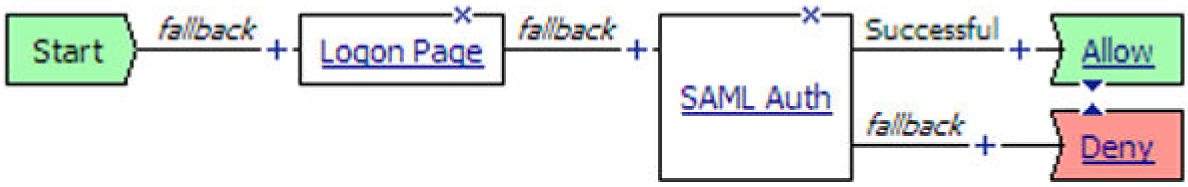
Scenario

You might bind multiple IdP connectors to an SP service on the BIG-IP system when you must provide services to different businesses and universities, each of which specifies an IdP to identify their users. When the user's information arrives at the SP service on the BIG-IP system, the SP service identifies the correct IdP and redirects the user to authenticate against that IdP before the SP service provides access to the service.

Note: *The SP service performs IdP discovery for a user only when the user initiates connection from an SP.*

Session variables and the typical access policy for BIG-IP system as SP

On a BIG-IP system configured as an SP, the typical access policy presents a logon page to the user. The Logon Page action populates session variables. You can customize the Logon Page action and affect session variable values. A SAML Auth action follows the logon page.



A SAML Auth action specifies an SP service. An SP service is an AAA service that requests authentication from an external IdP (specified in an IdP connector).

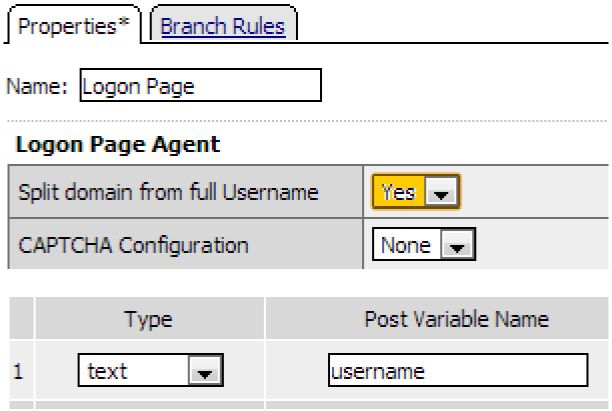
Session variables and SAML IdP discovery

Among multiple IdP connectors, the BIG-IP system must discover the correct external IdP with which to authenticate a user. For IdP discovery to work, you must specify matching criteria, a session variable name and value, for each IdP connector.

For example, users of a service might go to a particular landing page. When you bind the IdP connector, for the external IdP that serves those users, to the SP service, select the `%{session.server.landinguri}` session variable and supply a landing path value, such as, `/south*`. For users going to URLs such as `https://sp-service/southwest` and `https://sp-service/southeast`, the SP service selects the same IdP to authenticate them.

Logon Page action customization

These are some common customization examples for the Logon Page action.



Properties* | Branch Rules

Name: Logon Page

Logon Page Agent

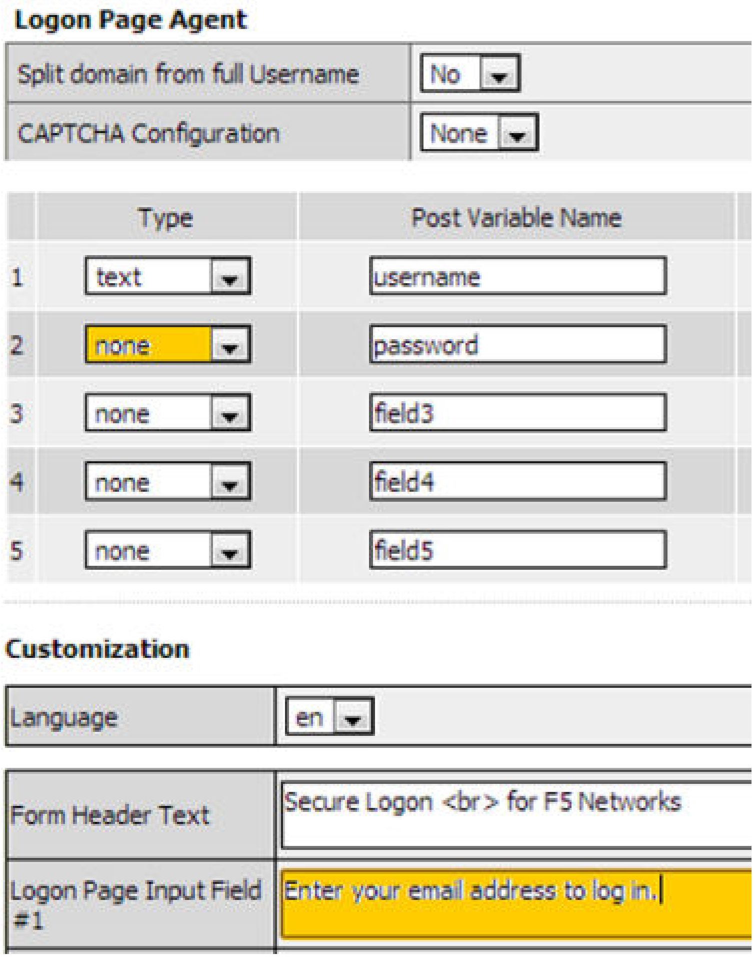
Split domain from full Username	Yes
CAPTCHA Configuration	None

	Type	Post Variable Name
1	text	username

Figure 28: Setting the value of session.logon.last.domain variable to the domain name only

Select **Yes** for **Split domain from full Username**. The Logon Page agent takes the user name, such as joe@office.com, that was entered and creates the following session variables with these values.

Session Variable	Value
%{session.logon.last.username}	joe
%{session.logon.last.domain}	office.com
%{session.logon.last.logonname}	joe@office.com



Logon Page Agent

Split domain from full Username	No
CAPTCHA Configuration	None

	Type	Post Variable Name
1	text	username
2	none	password
3	none	field3
4	none	field4
5	none	field5

Customization

Language: en

Form Header Text: Secure Logon <br> for F5 Networks

Logon Page Input Field #1: Enter your email address to log in.

Figure 29: Obtaining and email address as the username

Change the prompt for the first text field (username field). To omit the password: for **Type**, select **none** from the list.

About local IdP service

A *SAML IdP service* is a type of single sign-on (SSO) authentication service in Access Policy Manager® (APM®). When you use a BIG-IP® system as a SAML identity provider (IdP), a SAML IdP service provides SSO authentication for external SAML service providers (SPs). You must bind a SAML IdP service to SAML SP connectors, each of which specifies an external SP. APM responds to authentication requests from the service providers and produces assertions for them.

About SP connectors

A SAML service provider connector (an SP connector) specifies how a BIG-IP® system, configured as a SAML Identity Provider (IdP), connects with an external service provider.

What are the available ways I can configure a SAML SP connector?

You can use one or more of these methods to configure SAML service provider (SP) connectors in Access Policy Manager®.

- From metadata - Obtain a metadata file from the vendor and import it into Access Policy Manager. The advantage to this method is that the vendor provides the majority of all required data, including certificates. You can complete the configuration by simply typing a unique name for the SP connector, a very few additional required fields, and browsing to and importing the file. Access Policy Manager then configures the SP connector.

- From template - Use templates that Access Policy Manager provides for some vendors; for example, Google. The advantages to this method are that:
  - Most required data is included in the template
  - Additional required data is minimal. You can obtain it and certificates from the vendor

  After you select a template and type data into a few fields, Access Policy Manager configures the SP connector.
- Custom - Obtain information from the vendor and type the settings into the Configuration utility. To use this method, you must also obtain certificates from the vendor and import them into the BIG-IP® system. Use this method when a metadata file or a template for an SP connector is not available.

About local SP service

A *SAML SP service* is a type of AAA service in Access Policy Manager® (APM®). It requests authentication from an external SAML Identity Provider (IdP) that is specified on APM in a SAML IdP connector. (You bind a SAML service provider (SP) service to one or more SAML IdP connectors.) APM requests authentication from an IdP and consumes assertions from it to allow access to resources behind APM.

About IdP connectors

An IdP connector specifies how a BIG-IP® system, configured as a SAML service provider (SP), connects with an external SAML identity provider (IdP).

About methods for configuring SAML IdP connectors in APM

You can use one or more of these methods to configure SAML identity provider (IdP) connectors in Access Policy Manager® (APM®).

- From metadata - Obtain a metadata file from the vendor and import it into APM. The advantage to this method is that the vendor provides all required data, including the certificate. You can complete the configuration by simply typing a unique name for the identity provider, and browsing to and importing the file. APM imports the certificate to the BIG-IP® system and configures the SAML IdP connector.
- From template - Use templates that APM provides for some vendors. The advantages to this method are that:
  - Most required data is included in the template. (Note that the certificate is not included.)
  - Additional required data is minimal and is available from the vendor.

  APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- Custom - Research the identity provider requirements and type all settings into the Configuration utility. Use this method when a metadata file or a template for an identity provider is not available. APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- IdP Automation - Provide files with cumulative IdP metadata on remote systems, then configure BIG-IP IdP automation to poll the files periodically and create IdP connectors and bind them to a specific service provider (SP) service.

Task summary

Setting up SAML federation for BIG-IP® systems involves three major activities:

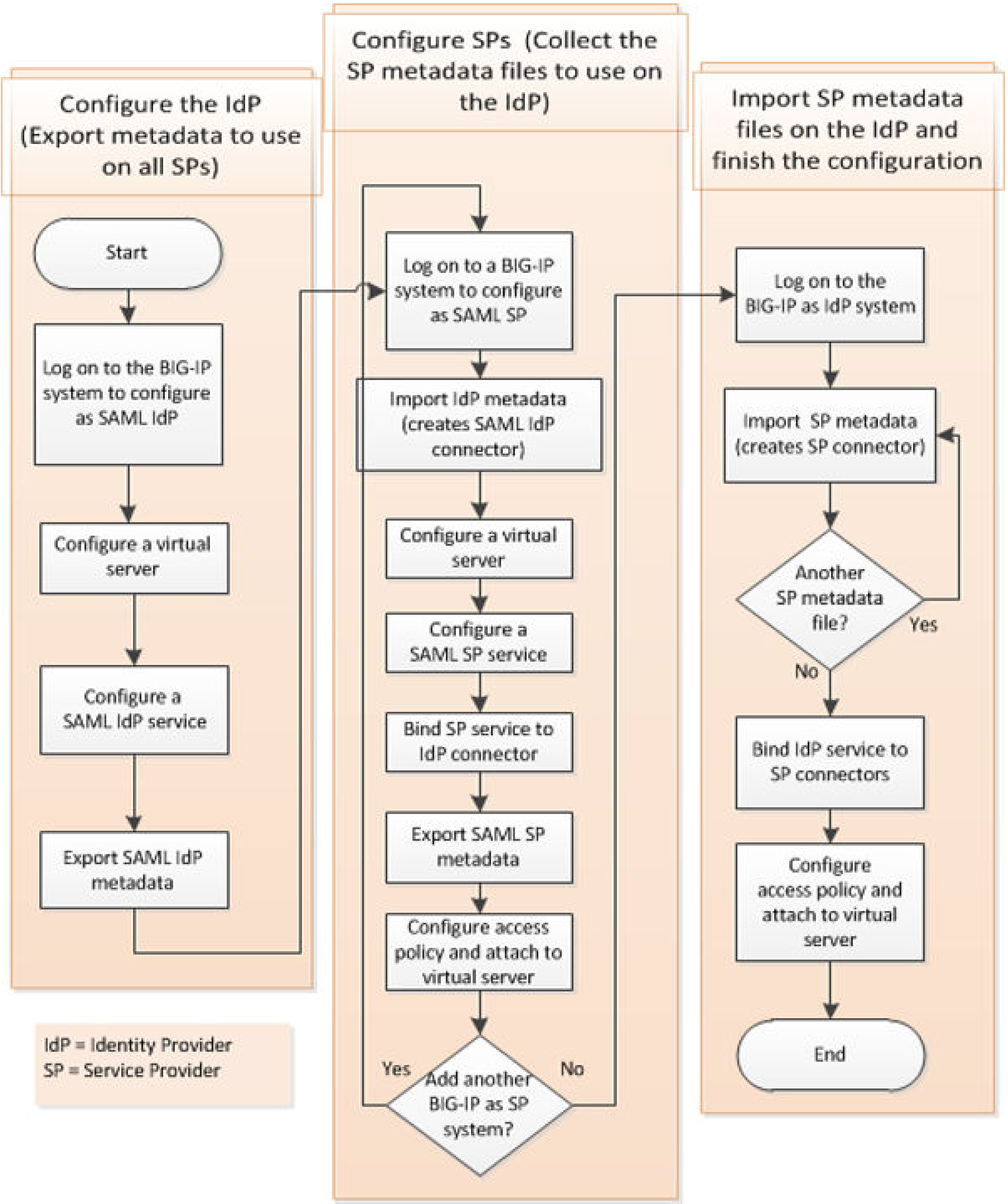
- First, you set up one BIG-IP system as a SAML identity provider (IdP) system
- Next, you set up one or more BIG-IP systems as a SAML service provider (SP)
- Last, you go back to the IdP system and set up connectivity to the SP systems

Task list

Setting up a BIG-IP system as a SAML IdP
Setting up a BIG-IP system as a SAML service provider system
Setting up connectivity from the IdP system to the SP systems

Flowchart: BIG-IP system federation configuration

This flowchart illustrates the process for configuring BIG-IP® systems in federation without providing an SSO portal.


Configure the IdP (Export metadata to use on all SPs)
Start
Log on to the BIG-IP system to configure as SAML IdP
Configure a virtual server
Configure a SAML IdP service
Export SAML IdP metadata
Configure SPs (Collect the SP metadata files to use on the IdP)
Log on to a BIG-IP system to configure as SAML SP
Import IdP metadata (creates SAML IdP connector)
Configure a virtual server
Configure a SAML SP service
Bind SP service to IdP connector
Export SAML SP metadata
Configure access policy and attach to virtual server
Add another BIG-IP as SP system?
Yes
No
Import SP metadata files on the IdP and finish the configuration
Log on to the BIG-IP as IdP system
Import SP metadata (creates SP connector)
Another SP metadata file?
Yes
No
Bind IdP service to SP connectors
Configure access policy and attach to virtual server
End
IdP = Identity Provider
SP = Service Provider

Setting up a BIG-IP system as a SAML IdP

You log in to the BIG-IP® system that you have selected to act as the SAML Identity Provider (IdP) so that you can configure elements that are required for SAML federation.

Log on to the BIG-IP system that you have selected to act as the SAML IdP in a SAML federation of BIG-IP systems.

Creating a virtual server for a BIG-IP (as SAML IdP) system

Before you start this task, configure a client SSL profile and a server SSL profile if you are going to create an SSL virtual server.

***Note:** Access Policy Manager® supports using a non-SSL virtual server for the BIG-IP® system configured as a SAML Identity Provider (IdP). However, we recommend using an SSL virtual server for security reasons. The following procedures include steps that are required for configuring an SSL virtual server, such as selecting client and server SSL profiles, and setting the service port to HTTPS.*

Specify a host virtual server to use as the SAML IdP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an IdP now appears on the Virtual Server List. The virtual server destination is available for use in one or more SAML IdP service configurations.

Configuring an artifact resolution service

Before you configure the artifact resolution service (ARS), you need to have configured a virtual server. That virtual server can be the same as the one used for the SAML Identity Provider (IdP), or you can create an additional virtual server.

***Note:** F5® highly recommends that the virtual server definition include a server SSL profile.*

You configure an ARS so that a BIG-IP® system that is configured as a SAML IdP can provide SAML artifacts in place of assertions. With ARS, the BIG-IP system can receive Artifact Resolve Requests (ARRQ) from service providers, and provide Artifact Resolve Responses (ARRP) for them.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **Artifact Resolution Services**.
2. Click **Create**.
   The Create New SAML Artifact Resolution Service popup screen opens, showing general settings.
3. In the **Name** field, type a name for the artifact resolution service.
4. In the **Description** field, type a new description.
5. Click **Service Settings.**
6. From the **Virtual Server** list, select the virtual server that you created previously.

   ARS listens on the IP address and port configured on the virtual server.

7. In the **Artifact Validity (Seconds)** field, type the number of seconds for which the artifact remains valid. The default is 60 seconds.

   The BIG-IP® system deletes the artifact if the number of seconds exceeds the artifact validity number.
8. For the **Send Method** setting, select the binding to use to send the artifact, either **POST** or **Redirect**.
9. In the **Host** field, type the host name defined for the virtual server, for example **ars.siterequest.com**.
10. In the **Port** field, type the port number defined in the virtual server. The default is `443`.
11. Click **Security Settings.**
12. To require that artifact resolution messages from an SP be signed, select the **Sign Artifact Resolution Request** check box.
13. To use HTTP Basic authentication for artifact resolution request messages, in the **User Name** field, type a name for the artifact resolution service request and in the **Password** field, type a password.

    These credentials must be present in all Artifact Resolve Requests sent to this ARS.
14. Click **OK**.
    The popup screen closes, leaving the Artifact Resolution Services list screen open.

The Artifact Resolution Service is ready for use.

Configuring a SAML IdP service

Configure a SAML Identity Provider (IdP) service for the BIG-IP® system, configured as a SAML IdP, to provide authentication service for SAML service providers (SPs).

***Note:** Configure this IdP service to meet the requirements of all SAML service providers that you bind with it.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Click **Create**.
   The Create New IdP Service popup screen displays.
3. In the **IdP Service Name** field, type a unique name for the SAML IdP service.
4. In the **IdP Entity ID** field, type a unique identifier for the IdP (this BIG-IP® system).

   Typically, the ID is a URI that points to the BIG-IP virtual server that is going to act as a SAML IdP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://siterequest.com/idp`, where the path points to the virtual server you use for BIG-IP system as a SAML IdP.
5. If the **IdP Entity ID** field does not contain a valid URI, you must provide one in the IdP Name Settings area:
   a) From the **Scheme** list select **https** or **http**.
   b) In the **Host** field, type a host name.
      For example, type `siterequest.com` in the **Host** field.
6. From the **Log Setting** list, select one of the following options:
   - Select an existing APM log setting.
   - Click **Create** to create a new log setting.
7.
8. If you select **SAML Profiles** on the left pane, the **Web Browser SSO** check box is selected by default.

   At least one profile must be selected.
9. To specify that this IdP use an artifact resolution service, click **Endpoint Settings** on the left pane and select a service from the **Artifact Resolution Service** list.
10. On the left pane, select **Assertion Settings** and complete the settings that display:

a) From the **Assertion Subject Type** list, select the type of subject for the IdP to authenticate.
b) From the **Assertion Subject Value** list, select the name of a session variable.

   This variable, *`%{session.logon.last.username}`*, is generally applicable. Some session variables are applicable depending on the type of authentication that you use for your site.
c) In the **Authentication Context Class Reference** field, select a URI reference.

   The URI reference identifies an authentication context class that describes an authentication context declaration.
d) In the **Assertion Validity (in seconds)** field, type the number of seconds for which the assertion is valid.
e) To encrypt the subject, select the **Enable encryption of Subject** check box.
   The **Encryption Strength** list becomes available.
f) From the **Encryption Strength** list, select a value.

   Supported values are AES128, AES192, and AES256.

11. On the left pane, select **SAML Attributes**, and for each attribute that you want to include in the attribute statement, repeat these substeps.
    a) Click **Add**.
       A Create New SAML Attribute popup screen displays.
    b) In the **Name** field, type a unique name for the attribute.

       Usually, the name is a fixed string, but it can be a session variable.
    c) To add a value to the attribute, click **Add**, type a value in the **Value(s)** field, and click **Update** to complete the addition.

       You can use a session variable for the value.

       This example shows using a fixed string for the name and a session variable for the value. Name: `user_telephonenumber` and value: `%{session.ad.last.attr.telephoneNumber}`.

       You can repeat this step to add multiple values for an attribute.
    d) To encrypt the values, select the **Encrypt** check box and select a value from the **Type** list.

       Supported values for type are AES128, AES192, and AES256.
    e) Click **OK**.
       The Create New SAML Attribute popup screen closes.
12. Click **Security Settings** from the left pane.
    a) From the **Signing Key** list, select the key from the BIG-IP system store.

       None is selected by default.
    b) From the **Signing Certificate** list, select the certificate from the BIG-IP system store.

       When selected, the IdP (the BIG-IP system) publishes this certificate to the service provider so the service provider can verify the assertion. **None** is selected by default.
13. Click **OK**.
    The popup screen closes. The new IdP service appears on the list.

Access Policy Manager® (APM®) creates a SAML IdP service. It is available to bind to SAML SP connectors. This service works with external service providers that share the same requirements for assertion settings and SAML attribute settings.

Exporting SAML IdP metadata from APM

You need to convey the SAML Identity Provider (IdP) metadata from Access Policy Manager® (APM®) to the external service providers that use the SAML IdP service. Exporting the IdP metadata for a SAML IdP service to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the table and click **Export Metadata**.

A popup screen opens, with **No** selected on the **Sign Metadata** list.

3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

      APM uses the key to sign the metadata.
   c) From the **Signature Verification Certificate** list, select a certificate.

      APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.
4. Select **OK**.
   APM downloads an XML file.

An XML file that contains IdP metadata is available.

Setting up a BIG-IP system as a SAML service provider system

You log in once to each BIG-IP® system that you have selected to act as a SAML service provider so that you can configure the elements on it that are required for federation with other BIG-IP systems, one of which functions as an SAML IdP.

Log on to a BIG-IP system that you have selected to act as a SAML SP in a federation of BIG-IP systems.

Configuring an IdP connector from IdP metadata

Locate the SAML IdP metadata file that you exported from the BIG-IP® system (as IdP). If the metadata file is signed, obtain the certificate also; import it into the BIG-IP system store on this device.

Import IdP metadata to create a SAML IdP connector on this BIG-IP system. The SAML IdP connector enables this BIG-IP system to connect and exchange information with the external BIG-IP system that acts as the IdP in the SAML federation.

1. On the menu bar, expand **SAML Service Provider** and click **External IdP Connectors**.
   The External IdP Connectors screen displays.
2. Select **Create** > **From Metadata**.
   The Create New SAML IdP Connector screen opens.
3. In the **Select File** field, browse to and select the metadata file for the IdP.
4. In the **Identity Provider Name** field, type a unique name for the IdP.
5. If the metadata is signed, select a certificate from the **Select Signing Certificate** list.
6. Click **OK**.
   The file is uploaded, the SAML IdP connector is created, and the screen closes.

The SAML IdP connector is displayed on the SAML IdP Connectors list.

Creating a virtual server for a BIG-IP (as SAML SP) system

Specify a host virtual server to use as the SAML SP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.

5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an SP now appears on the Virtual Server List. The virtual server destination is available for use in a SAML SP service configuration.

Configuring a SAML SP service for federation

Configure a SAML service provider (SP) service for Access Policy Manager® to provide AAA authentication, requesting authentication and receiving assertions from a SAML IdP.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Click **Create**.
   The Create New SAML SP Service screen opens.
3. In the **Name** field, type a unique name for the SAML SP service.
4. In the **Entity ID** field, type a unique identifier for the service provider.

   Typically entity ID is a URI that points to the BIG-IP virtual server that is going to act as SAML SP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://bigip-sp`, where `https:/bigip-sp` points to the virtual server you use for BIG-IP system as a SAML service provider.
5. If the **Entity ID** field does not contain a valid URI, in the SP Name Settings area from the **Scheme** list, select **https** or **http** and in the **Host** field, type a host name.
   For example, type `siterequest.com` in the **Host** field.
6. In the **Relay State** field, type a scheme, host, and path.

   This is a path is where this BIG-IP® system redirects users after they are authenticated.
7. For this service provider to request an artifact instead of an assertion from the IdP, from the left pane select **Endpoint Settings** and, from the **Assertion Consumer Service Binding** list, select **Artifact**.

   POST is the default setting.
8. From the left pane, select **Security Settings**.

   The screen displays the applicable settings.
9. If you want this BIG-IP system to send signed authentication requests to the SAML IdP, select **Signed Authentication Request**. Then select a key and a certificate from those in the BIG-IP system store from the **Message Signing Private Key** and **Message Signing Certificate** lists.
10. If this BIG-IP system requires signed assertions from the SAML IdP, ensure that the **Want Signed Assertion** check box remains selected.
11. If this BIG-IP system requires encrypted assertions from the SAML IdP, select **Want Encrypted Assertion**. Then select a key and a certificate from those in the BIG-IP system store from the **Assertion Decryption Private Key** and **Assertion Decryption Certificate** lists.

    The BIG-IP system uses the private key and certificate to decrypt the assertion.
12. To configure additional service provider attributes, from the left pane click **Advanced**.

    The screen displays the applicable settings.
13. To force users to authenticate again even when they have an SSO session at the identity provider, select the **Force Authentication** check box.

    This setting is for use when the external IdP supports a force authentication flag.

14. To allow the external IdP, when processing requests from this BIG-IP system as SP, to create a new identifier to represent the principal, select the **Allow Name-Identifier Creation** check box.
15. To specify the type of identifier information to use, select a URI reference from the **Name-Identifier Policy Format** list.
    For example, if a Service Provider (SP) initiates SSO by sending an `AuthnRequest` to the IdP with format **urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress**, then the IdP response should contain the subject identity in email format.
16. To specify that the assertion subject's identifier be returned in the namespace of an SP other than the requester, or in the namespace of a SAML affiliation group of SPs, type a value in the **SP Name-Identifier Qualifier** field.
17. Click **OK**.
    The screen closes.

APM® creates the SAML SP service. It is available to bind to SAML IdP connectors and to export to a metadata file.

Binding the BIG-IP system (as IdP) with the SP service on this device

Bind the SAML SP service for this device (BIG-IP® system) to the SAML IdP connector for the external BIG-IP system that acts as the IdP, so that this device requests authentication service from the IdP.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select a SAML SP service from the list.
3. Click **Bind/Unbind IdP Connectors**.
   A pop-up screen displays a list of any IdP connectors that are associated with this SP service.
4. Click **Add New Row**.
5. Select the SAML IdP connector for the BIG-IP system that acts as the IdP in the federation.

   Because you are binding only one IdP connector to the SP service, you do not need to fill in the **Matching Source** and **Matching Value** fields.
6. Click **Update**.

   The configuration is not saved until you click **OK**.
7. Click **OK**.
   APM® saves the configuration. The screen closes.

The SAML IdP connector that you selected is bound to the SAML SP service.

Exporting SAML SP metadata from APM

You need to convey the SP metadata from APM® to the external SAML IdP that provides authentication service to this SP. Exporting the SAML SP metadata to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select an SP service from the list and click **Export Metadata**.
   A popup window opens, displaying **No** on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

      APM uses the key to sign the metadata.
   c) From the **Signature Verification Certificate** list, select a certificate.

APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.

4. Select **OK**.
   APM downloads an XML file.

You must either import the XML file on the IdP system or use the information in the XML file to configure SP metadata on the IdP system .

Configuring an access policy to authenticate with an external SAML IdP

Before you start this task, configure an access profile.

When you use this BIG-IP® system as a SAML service provider (SP), configure an access policy to direct users to an external SAML Identity Provider (IdP) for authentication.

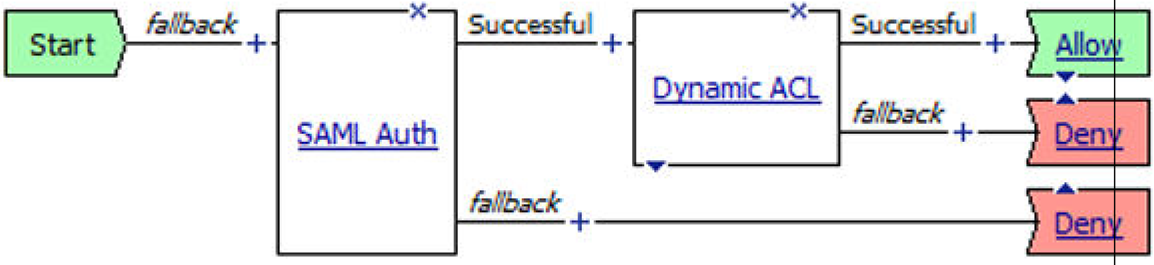
1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Authentication tab, select **SAML Auth** and click the **Add Item** button.
   The SAML Auth properties window opens.
5. In the SAML Authentication SP area from the **AAA Server** list, select a SAML SP service and click **Save**.
   The Access Policy window displays.
6. Add any additional actions that you require to complete the policy.
7. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
8. At the top of the window, click the **Apply Access Policy** link to apply and activate your changes to this access policy.
9. Click the **Close** button to close the visual policy editor.

You have an access policy that uses SAML authentication against an external SAML IdP and further qualifies the resources that a user can access.

Simple access policy to authenticate users against an external SAML IdP



To apply this access policy to network traffic, add the access profile to a virtual server.

__Note:__ To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

__Note:__ Log settings are configured in the __Access__ > __Overview__ > __Event Log__ > __Settings__ area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   __Note:__ Logging is disabled when the __Selected__ list is empty.
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Setting up connectivity from the IdP system to the SP systems

You log in to the BIG-IP® system that you configured as the SAML Identity Provider (IdP) so that you can set up connectivity to the BIG-IP systems you configured as SAML service providers (SPs).

Log on to the BIG-IP system that you have selected to act as the SAML IdP in a SAML federation of BIG-IP systems.

Configuring SAML SP connectors from SAML SP metadata files

Import SP metadata into this BIG-IP® system from each BIG-IP system that is configured as an SP to create SP connectors in this system that you can use to create a federation of BIG-IP systems.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **External SP Connectors**.

A list of SAML SP connectors displays.

2. Select **Create** > **From Metadata**
   The Create New SAML Service Provider window opens.
3. In the **Select File** field, browse to and select the metadata file for the service provider.
4. In the **Service Provider Name** field, type a unique name for the service provider.
5. If the metadata is signed, select the certificate from the **Select Signing Certificate** list.
6. Click **OK**.
   The file is uploaded, the SAML SP connector is created, and the window closes.

The SAML SP connector is displayed on the External SP Connectors list.

Binding IdP service and SP connectors for federation

Select a SAML Identity Provider (IdP) service and the SAML service provider (SP) connectors that use the service so that this BIG-IP® system can provide authentication (SAML IdP service) to external SAML service providers.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the list.

   A SAML IdP service provides authentication service.
3. Click **Bind/Unbind SP Connectors**.
   The screen displays a list of available SAML SP connectors.
4. Select the SAML SP connectors for the external BIG-IP systems that are configured as SPs and that you want to use this service.
5. Click **OK**.
   The screen closes.

The SAML IdP service is bound to the external SAML service providers specified in the SAML SP connectors.

Creating an access profile associated with the SAML IdP service

Use this procedure when this BIG-IP® system, as a SAML Identity Provider (IdP), supports service provider-initiated connections only.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*
4. In the SSO Across Authentication Domains (Single Domain mode) area, from the **SSO Configuration** list, select the name of the local SAML IdP service.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Configuring an access policy to provide authentication from the local IdP

Configure an access policy so that this BIG-IP® system, as a SAML Identity Provider (IdP) can provide authentication for SAML service providers.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

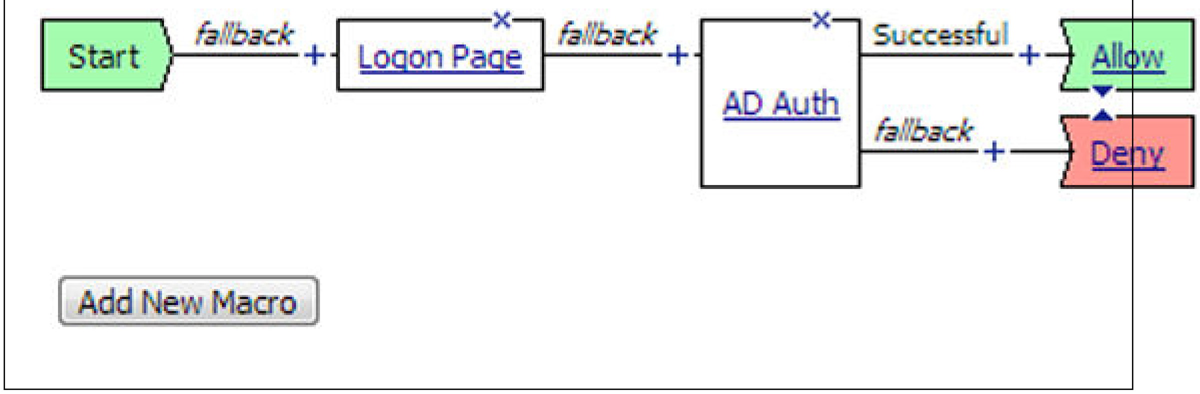
   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Add one or more authentication checks on the fallback branch after the **Logon Page** action.

   Select the authentication checks that are appropriate for application access at your site.
7. (Optional) Add any other branches and actions that you need to complete the policy.
8. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
9. Click the **Apply Access Policy** link to apply and activate the changes to the policy.
10. Click the **Close** button to close the visual policy editor.

You have an access policy that presents a logon page and authenticates the user..

Access policy to provide authentication for SAML service providers when this BIG-IP system is the IdP



To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

BIG-IP System Federation for SP- and IdP-Initiated Connections

Overview: Federating BIG-IP systems for SAML SSO (with an SSO portal)

In a federation of BIG-IP® systems, one BIG-IP system acts as a SAML Identity Provider (IdP) and other BIG-IP systems act as SAML service providers.

This configuration supports:

- Connections that initiate at the IdP or at SAML service providers.
- Service providers that require different types of subject, attributes, and security settings for assertions.

About local IdP service

A *SAML IdP service* is a type of single sign-on (SSO) authentication service in Access Policy Manager® (APM®). When you use a BIG-IP® system as a SAML identity provider (IdP), a SAML IdP service provides SSO authentication for external SAML service providers (SPs). You must bind a SAML IdP service to SAML SP connectors, each of which specifies an external SP. APM responds to authentication requests from the service providers and produces assertions for them.

About SP connectors

A SAML service provider connector (an SP connector) specifies how a BIG-IP® system, configured as a SAML Identity Provider (IdP), connects with an external service provider.

What are the available ways I can configure a SAML SP connector?

You can use one or more of these methods to configure SAML service provider (SP) connectors in Access Policy Manager®.

- From metadata - Obtain a metadata file from the vendor and import it into Access Policy Manager. The advantage to this method is that the vendor provides the majority of all required data, including certificates. You can complete the configuration by simply typing a unique name for the SP connector, a very few additional required fields, and browsing to and importing the file. Access Policy Manager then configures the SP connector.
- From template - Use templates that Access Policy Manager provides for some vendors; for example, Google. The advantages to this method are that:

  - Most required data is included in the template
  - Additional required data is minimal. You can obtain it and certificates from the vendor

  After you select a template and type data into a few fields, Access Policy Manager configures the SP connector.
- Custom - Obtain information from the vendor and type the settings into the Configuration utility. To use this method, you must also obtain certificates from the vendor and import them into the BIG-IP® system. Use this method when a metadata file or a template for an SP connector is not available.

About local SP service

A *SAML SP service* is a type of AAA service in Access Policy Manager® (APM®). It requests authentication from an external SAML Identity Provider (IdP) that is specified on APM in a SAML IdP connector. (You bind a SAML service provider (SP) service to one or more SAML IdP connectors.) APM requests authentication from an IdP and consumes assertions from it to allow access to resources behind APM.

About IdP connectors

An IdP connector specifies how a BIG-IP® system, configured as a SAML service provider (SP), connects with an external SAML identity provider (IdP).

About methods for configuring SAML IdP connectors in APM

You can use one or more of these methods to configure SAML identity provider (IdP) connectors in Access Policy Manager® (APM®).

- From metadata - Obtain a metadata file from the vendor and import it into APM. The advantage to this method is that the vendor provides all required data, including the certificate. You can complete the configuration by simply typing a unique name for the identity provider, and browsing to and importing the file. APM imports the certificate to the BIG-IP® system and configures the SAML IdP connector.
- From template - Use templates that APM provides for some vendors. The advantages to this method are that:

  - Most required data is included in the template. (Note that the certificate is not included.)
  - Additional required data is minimal and is available from the vendor.

  APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- Custom - Research the identity provider requirements and type all settings into the Configuration utility. Use this method when a metadata file or a template for an identity provider is not available. APM configures the SAML IdP connector. You must obtain a certificate from the vendor and import it into the BIG-IP system.
- IdP Automation - Provide files with cumulative IdP metadata on remote systems, then configure BIG-IP IdP automation to poll the files periodically and create IdP connectors and bind them to a specific service provider (SP) service.

Task summary

Setting up SAML federation for BIG-IP® systems involves three major activities:

- First, you set up one BIG-IP system as a SAML identity provider (IdP) system
- Next, you set up one or more BIG-IP systems as a SAML service provider (SP)
- Last, you go back to the IdP system and set up connectivity to the SP systems

Task list

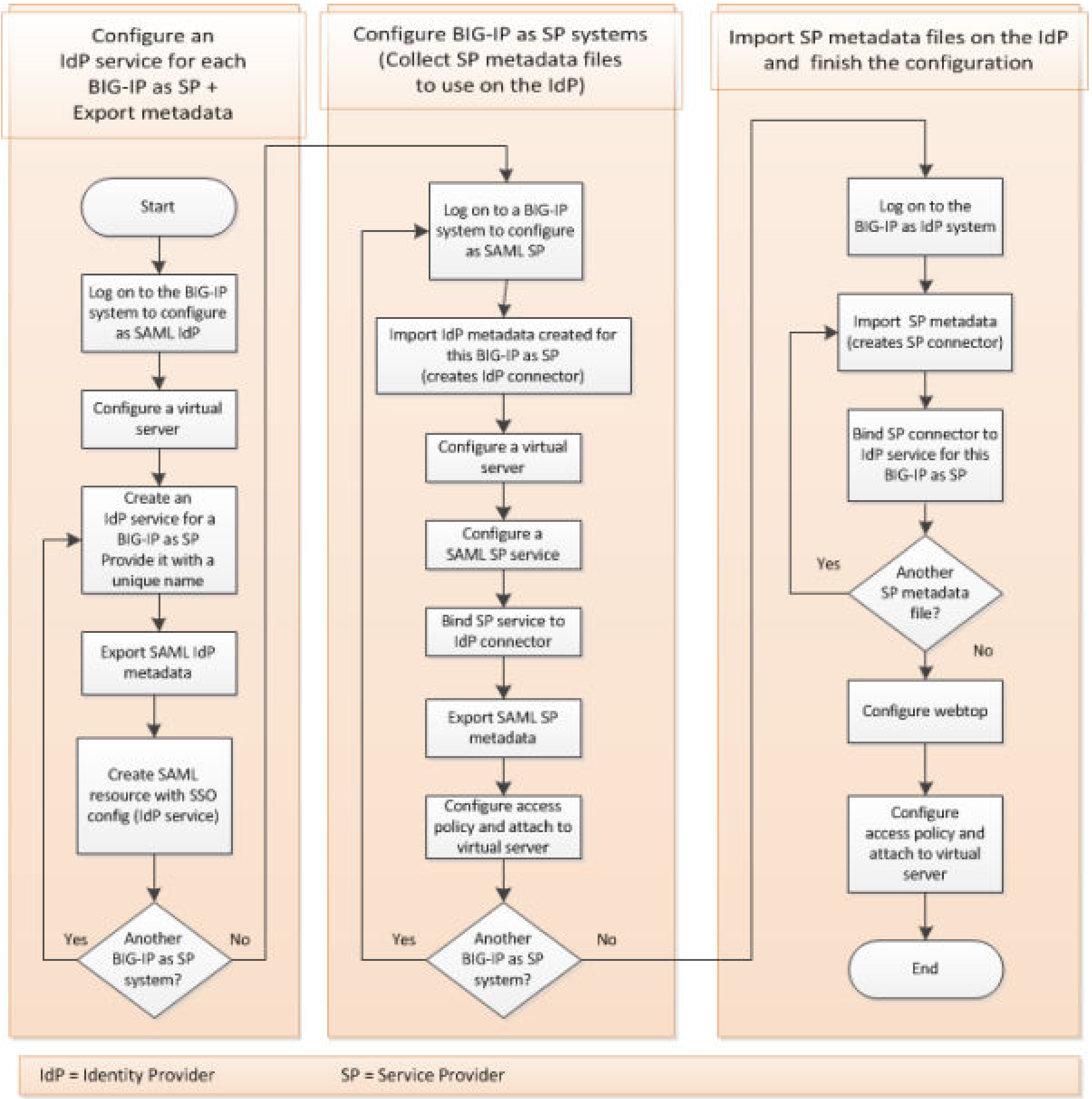
Setting up a BIG-IP system as a SAML IdP

Setting up a BIG-IP system as a SAML service provider system

Setting up connectivity from the IdP system to the SP systems

Flowchart: BIG-IP system federation configuration with SSO portal

This flowchart illustrates the process for configuring BIG-IP® systems in federation and providing an SSO portal.


Configure an IdP service for each BIG-IP as SP + Export metadata
Configure BIG-IP as SP systems (Collect SP metadata files to use on the IdP)
Import SP metadata files on the IdP and finish the configuration
Start
Log on to the BIG-IP system to configure as SAML IdP
Configure a virtual server
Create an IdP service for a BIG-IP as SP Provide it with a unique name
Export SAML IdP metadata
Create SAML resource with SSO config (IdP service)
Another BIG-IP as SP system?
Yes
No
Log on to a BIG-IP system to configure as SAML SP
Import IdP metadata created for this BIG-IP as SP (creates IdP connector)
Configure a virtual server
Configure a SAML SP service
Bind SP service to IdP connector
Export SAML SP metadata
Configure access policy and attach to virtual server
Another BIG-IP as SP system?
Yes
No
Log on to the BIG-IP as IdP system
Import SP metadata (creates SP connector)
Bind SP connector to IdP service for this BIG-IP as SP
Another SP metadata file?
Yes
No
Configure webtop
Configure access policy and attach to virtual server
End
IdP = Identity Provider
SP = Service Provider

Setting up a BIG-IP system as a SAML IdP

You log in to the BIG-IP® system that you have selected to act as the SAML Identity Provider (IdP) so that you can configure elements that are required for SAML federation.

Log on to the BIG-IP system that you have selected to act as the SAML IdP in a SAML federation of BIG-IP systems.

Creating a virtual server for a BIG-IP (as SAML IdP) system

Before you start this task, configure a client SSL profile and a server SSL profile if you are going to create an SSL virtual server.

***Note:** Access Policy Manager® supports using a non-SSL virtual server for the BIG-IP® system configured as a SAML Identity Provider (IdP). However, we recommend using an SSL virtual server for security reasons. The following procedures include steps that are required for configuring an SSL virtual server, such as selecting client and server SSL profiles, and setting the service port to HTTPS.*

Specify a host virtual server to use as the SAML IdP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an IdP now appears on the Virtual Server List. The virtual server destination is available for use in one or more SAML IdP service configurations.

Configuring an artifact resolution service for BIG-IP federation with SSO portal

Before you configure the artifact resolution service (ARS), you need to have configured a virtual server. That virtual server can be the same as one that is used for an SAML Identity Provider (IdP), or you can create an additional virtual server.

***Note:** F5® recommends that the virtual server definition include a server SSL profile.*

You configure an ARS so that a BIG-IP® system that is configured as a SAML IdP can provide SAML artifacts in place of assertions. With ARS, the BIG-IP system can receive Artifact Resolve Requests (ARRQ) from service providers, and provide Artifact Resolve Responses (ARRP) for them. You can configure one or more ARS.

***Note:** In a BIG-IP system federation configuration that supports an SSO portal, you must configure one IdP service for each Service Provider. However, you do not need to configure a separate ARS for each IdP service.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **Artifact Resolution Services**.
2. Click **Create**.
   The Create New SAML Artifact Resolution Service popup screen opens, showing general settings.
3. In the **Name** field, type a name for the artifact resolution service.
4. In the **Description** field, type a new description.
5. Click **Service Settings**.
6. From the **Virtual Server** list, select the virtual server that you created previously.

   ARS listens on the IP address and port configured on the virtual server.
7. In the **Artifact Validity (Seconds)** field, type the number of seconds for which the artifact remains valid. The default is `60` seconds.

   The BIG-IP® system deletes the artifact if the number of seconds exceeds the artifact validity number.
8. For the **Send Method** setting, select the binding to use to send the artifact, either **POST** or **Redirect**.
9. In the **Host** field, type the host name defined for the virtual server, for example **ars.siterequest.com**.

10. In the **Port** field, type the port number defined in the virtual server. The default is `443`.
11. Click **Security Settings**.
12. To require that artifact resolution messages from an SP be signed, select the **Sign Artifact Resolution Request** check box.
13. To use HTTP Basic authentication for artifact resolution request messages, in the **User Name** field, type a name for the artifact resolution service request and in the **Password** field, type a password.

    These credentials must be present in all Artifact Resolve Requests sent to this ARS.
14. Click **OK**.
    The popup screen closes, leaving the Artifact Resolution Services list screen open.

The Artifact Resolution Service is ready for use.

Configuring a SAML IdP service for one SP connector

Configure a SAML Identity Provider (IdP) service for Access Policy Manager®, as a SAML IdP, to provide single sign-on authentication for one SAML service provider (SP).

***Note:** Configure one IdP service for each SAML service provider.*

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Click **Create**.
   The Create New IdP Service popup screen displays.
3. In the **IdP Service Name** field, type a unique name for the SAML IdP service.
4. In the **IdP Entity ID** field, type a unique identifier for the IdP (this BIG-IP® system).

   Typically, the ID is a URI that points to the BIG-IP virtual server that is going to act as a SAML IdP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://siterequest.com/idp`, where the path points to the virtual server you use for BIG-IP system as a SAML IdP.
5. If the **IdP Entity ID** field does not contain a valid URI, you must provide one in the IdP Name Settings area:
   a) From the **Scheme** list select **https** or **http**.
   b) In the **Host** field, type a host name.
      For example, type `siterequest.com` in the **Host** field.
6. If you select **SAML Profiles** on the left pane, the **Web Browser SSO** check box is selected by default.

   At least one profile must be selected.
7. To specify that this IdP use an artifact resolution service, click **Endpoint Settings** on the left pane and select a service from the **Artifact Resolution Service** list.
8. On the left pane, select **Assertion Settings** and complete the settings that display:
   a) From the **Assertion Subject Type** list, select the type of subject for the IdP to authenticate.
   b) From the **Assertion Subject Value** list, select the name of a session variable.

      This variable, *`%{session.logon.last.username}`*, is generally applicable. Some session variables are applicable depending on the type of authentication that you use for your site.
   c) In the **Authentication Context Class Reference** field, select a URI reference.

      The URI reference identifies an authentication context class that describes an authentication context declaration.
   d) In the **Assertion Validity (in seconds)** field, type the number of seconds for which the assertion is valid.
   e) To encrypt the subject, select the **Enable encryption of Subject** check box.

The **Encryption Strength** list becomes available.

f) From the **Encryption Strength** list, select a value.

Supported values are AES128, AES192, and AES256.

9. On the left pane, select **SAML Attributes**, and for each attribute that you want to include in the attribute statement, repeat these substeps.

   a) Click **Add**.
   A Create New SAML Attribute popup screen displays.

   b) In the **Name** field, type a unique name for the attribute.

   Usually, the name is a fixed string, but it can be a session variable.

   c) To add a value to the attribute, click **Add**, type a value in the **Value(s)** field, and click **Update** to complete the addition.

   You can use a session variable for the value.

   This example shows using a fixed string for the name and a session variable for the value. Name: `user_telephonenumber` and value: `%{session.ad.last.attr.telephoneNumber}`.

   You can repeat this step to add multiple values for an attribute.

   d) To encrypt the values, select the **Encrypt** check box and select a value from the **Type** list.

   Supported values for type are AES128, AES192, and AES256.

   e) Click **OK**.
   The Create New SAML Attribute popup screen closes.

10. Click **Security Settings** from the left pane.

    a) From the **Signing Key** list, select the key from the BIG-IP system store.

    None is selected by default.

    b) From the **Signing Certificate** list, select the certificate from the BIG-IP system store.

    When selected, the IdP (the BIG-IP system) publishes this certificate to the service provider so the service provider can verify the assertion. **None** is selected by default.

11. Click **OK**.
    The popup screen closes. The new IdP service appears on the list.

Access Policy Manager® (APM®) creates a SAML IdP service. It is available to bind to an SP connector.

Exporting SAML IdP metadata from APM

You need to convey the SAML Identity Provider (IdP) metadata from Access Policy Manager® (APM®) to the external service providers that use the SAML IdP service. Exporting the IdP metadata for a SAML IdP service to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the table and click **Export Metadata**.
   A popup screen opens, with **No** selected on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:

   a) From the **Sign Metadata** list, select **Yes**.

   b) From the **Signing Key** list, select a key.

   APM uses the key to sign the metadata.

   c) From the **Signature Verification Certificate** list, select a certificate.

   APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.

4. Select **OK**.
   APM downloads an XML file.

An XML file that contains IdP metadata is available.

Configuring a SAML resource and attaching a SAML IdP service

Configure a SAML resource to provide access to services on a SAML service provider when using Access Policy Manager® (APM®) as a SAML IdP.

***Note:** Configure one SAML resource for each SAML IdP service that you have configured.*

1. On the Main tab, click **Access** > **Federation** > **SAML Resources**.
   The SAML Resources list screen opens.
2. Click the **Create** button.
   The SAML Resource New Resource window opens
3. In the **Name** field, type a unique name for the SAML resource.
4. Do not clear the **Publish on Webtop** check box unless when you want to remove this resource from the webtop.

   When **Publish on Webtop** is selected, the SAML resource is displayed on a webtop where a user can initiate connection to an SP by clicking the icon. If you want users to initiate connection to this resource from an external SAML service provider only and do not want to show this resource on a webtop, clear the check box.
5. In the Configuration area from the **SSO Configuration** list, select the SAML IdP service that is bound to the SAML SP connector with the resources you want.
6. In the **Customization Settings for English** area in the **Caption** field, type a caption for this SAML resource.
7. Click **Finished**.
   The SAML resource is created and associated with a SAML IdP service that is bound to one external service provider.

Setting up a BIG-IP system as a SAML service provider system

You log in once to each BIG-IP® system that you have selected to act as a SAML service provider so that you can configure the elements on it that are required for federation with other BIG-IP systems, one of which functions as an SAML IdP.

Log on to a BIG-IP system that you have selected to act as a SAML SP in a federation of BIG-IP systems.

Configuring an IdP connector from IdP metadata

Locate the SAML IdP metadata file that you exported from the BIG-IP® system (as IdP). If the metadata file is signed, obtain the certificate also; import it into the BIG-IP system store on this device.

Import IdP metadata to create a SAML IdP connector on this BIG-IP system. The SAML IdP connector enables this BIG-IP system to connect and exchange information with the external BIG-IP system that acts as the IdP in the SAML federation.

1. On the menu bar, expand **SAML Service Provider** and click **External IdP Connectors**.
   The External IdP Connectors screen displays.
2. Select **Create** > **From Metadata**.
   The Create New SAML IdP Connector screen opens.
3. In the **Select File** field, browse to and select the metadata file for the IdP.
4. In the **Identity Provider Name** field, type a unique name for the IdP.
5. If the metadata is signed, select a certificate from the **Select Signing Certificate** list.
6. Click **OK**.
   The file is uploaded, the SAML IdP connector is created, and the screen closes.

The SAML IdP connector is displayed on the SAML IdP Connectors list.

Creating a virtual server for a BIG-IP (as SAML SP) system

Specify a host virtual server to use as the SAML SP.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address for a host virtual server.

   This field accepts an address in CIDR format (IP address/prefix). However, when you type the complete IP address for a host, you do not need to type a prefix after the address.
5. In the **Service Port** field, type `443` or select **HTTPS** from the list.
6. For the **HTTP Profile** setting, verify that the default HTTP profile, **http**, is selected.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created and move the name to the **Selected** list.
8. For the **SSL Profile (Server)** setting, from the **Available** list, select the name of the Server SSL profile you previously created and move the name to the **Selected** list.
9. Click **Finished**.

The virtual server for the BIG-IP system configured as an SP now appears on the Virtual Server List. The virtual server destination is available for use in a SAML SP service configuration.

Configuring a SAML SP service for federation

Configure a SAML service provider (SP) service for Access Policy Manager® to provide AAA authentication, requesting authentication and receiving assertions from a SAML IdP.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Click **Create**.
   The Create New SAML SP Service screen opens.
3. In the **Name** field, type a unique name for the SAML SP service.
4. In the **Entity ID** field, type a unique identifier for the service provider.

   Typically entity ID is a URI that points to the BIG-IP virtual server that is going to act as SAML SP. If the entity ID is not a valid URL, the **Host** field is required.

   For example, type `https://bigip-sp`, where `https:/bigip-sp` points to the virtual server you use for BIG-IP system as a SAML service provider.
5. If the **Entity ID** field does not contain a valid URI, in the SP Name Settings area from the **Scheme** list, select **https** or **http** and in the **Host** field, type a host name.
   For example, type `siterequest.com` in the **Host** field.
6. In the **Relay State** field, type a scheme, host, and path.

   This is a path is where this BIG-IP® system redirects users after they are authenticated.
7. For this service provider to request an artifact instead of an assertion from the IdP, from the left pane select **Endpoint Settings** and, from the **Assertion Consumer Service Binding** list, select **Artifact**.

   POST is the default setting.
8. From the left pane, select **Security Settings**.

   The screen displays the applicable settings.
9. If you want this BIG-IP system to send signed authentication requests to the SAML IdP, select **Signed Authentication Request**. Then select a key and a certificate from those in the BIG-IP system store from the **Message Signing Private Key** and **Message Signing Certificate** lists.

10. If this BIG-IP system requires signed assertions from the SAML IdP, ensure that the **Want Signed Assertion** check box remains selected.
11. If this BIG-IP system requires encrypted assertions from the SAML IdP, select **Want Encrypted Assertion**. Then select a key and a certificate from those in the BIG-IP system store from the **Assertion Decryption Private Key** and **Assertion Decryption Certificate** lists.

    The BIG-IP system uses the private key and certificate to decrypt the assertion.
12. To configure additional service provider attributes, from the left pane click **Advanced**.

    The screen displays the applicable settings.
13. To force users to authenticate again even when they have an SSO session at the identity provider, select the **Force Authentication** check box.

    This setting is for use when the external IdP supports a force authentication flag.
14. To allow the external IdP, when processing requests from this BIG-IP system as SP, to create a new identifier to represent the principal, select the **Allow Name-Identifier Creation** check box.
15. To specify the type of identifier information to use, select a URI reference from the **Name-Identifier Policy Format** list.
    For example, if a Service Provider (SP) initiates SSO by sending an `AuthnRequest` to the IdP with format **urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress**, then the IdP response should contain the subject identity in email format.
16. To specify that the assertion subject's identifier be returned in the namespace of an SP other than the requester, or in the namespace of a SAML affiliation group of SPs, type a value in the **SP Name-Identifier Qualifier** field.
17. Click **OK**.
    The screen closes.

APM® creates the SAML SP service. It is available to bind to SAML IdP connectors and to export to a metadata file.

Binding the BIG-IP system (as IdP) with the SP service on this device

Bind the SAML SP service for this device (BIG-IP® system) to the SAML IdP connector for the external BIG-IP system that acts as the IdP, so that this device requests authentication service from the IdP.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select a SAML SP service from the list.
3. Click **Bind/Unbind IdP Connectors**.
   A pop-up screen displays a list of any IdP connectors that are associated with this SP service.
4. Click **Add New Row**.
5. Select the SAML IdP connector for the BIG-IP system that acts as the IdP in the federation.

   Because you are binding only one IdP connector to the SP service, you do not need to fill in the **Matching Source** and **Matching Value** fields.
6. Click **Update**.

   The configuration is not saved until you click **OK**.
7. Click **OK**.

   APM® saves the configuration. The screen closes.

The SAML IdP connector that you selected is bound to the SAML SP service.

Exporting SAML SP metadata from APM

You need to convey the SP metadata from APM® to the external SAML IdP that provides authentication service to this SP. Exporting the SAML SP metadata to a file provides you with the information that you need to do this.

1. On the Main tab, click **Access** > **Federation** > **SAML Service Provider**.
   The Local SP Services screen displays.
2. Select an SP service from the list and click **Export Metadata**.
   A popup window opens, displaying **No** on the **Sign Metadata** list.
3. For APM to sign the metadata, perform these steps:
   a) From the **Sign Metadata** list, select **Yes**.
   b) From the **Signing Key** list, select a key.

      APM uses the key to sign the metadata.
   c) From the **Signature Verification Certificate** list, select a certificate.

      APM exports the certificate to the metadata file. The system on which you import the metadata file can use the certificate to verify the metadata signature.
4. Select **OK**.
   APM downloads an XML file.

You must either import the XML file on the IdP system or use the information in the XML file to configure SP metadata on the IdP system .

Configuring an access policy to authenticate with an external SAML IdP

Before you start this task, configure an access profile.

When you use this BIG-IP® system as a SAML service provider (SP), configure an access policy to direct users to an external SAML Identity Provider (IdP) for authentication.

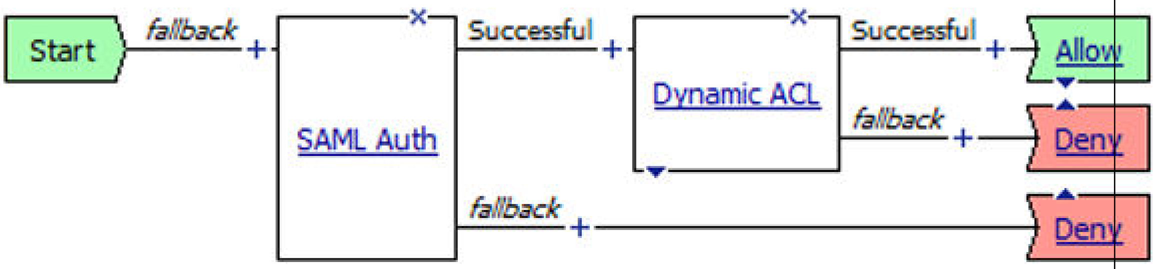
1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Authentication tab, select **SAML Auth** and click the **Add Item** button.
   The SAML Auth properties window opens.
5. In the SAML Authentication SP area from the **AAA Server** list, select a SAML SP service and click **Save**.
   The Access Policy window displays.
6. Add any additional actions that you require to complete the policy.
7. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
8. At the top of the window, click the **Apply Access Policy** link to apply and activate your changes to this access policy.
9. Click the **Close** button to close the visual policy editor.

You have an access policy that uses SAML authentication against an external SAML IdP and further qualifies the resources that a user can access.

Simple access policy to authenticate users against an external SAML IdP



To apply this access policy to network traffic, add the access profile to a virtual server.

Note: *To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

Note: *Log settings are configured in the* ***Access*** > ***Overview*** > ***Event Log*** > ***Settings*** *area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. On the menu bar, click **Logs**.
   The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   Note: *Logging is disabled when the* ***Selected*** *list is empty.*
5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

Setting up connectivity from the IdP system to the SP systems

You log in to the BIG-IP® system that you configured as the SAML Identity Provider (IdP) so that you can set up connectivity to the BIG-IP systems you configured as SAML service providers (SPs).

Log on to the BIG-IP system that you have selected to act as the SAML IdP in a SAML federation of BIG-IP systems.

Configuring SAML SP connectors from SAML SP metadata files

Import SP metadata into this BIG-IP® system from each BIG-IP system that is configured as an SP to create SP connectors in this system that you can use to create a federation of BIG-IP systems.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider** > **External SP Connectors**.
   A list of SAML SP connectors displays.
2. Select **Create** > **From Metadata**
   The Create New SAML Service Provider window opens.
3. In the **Select File** field, browse to and select the metadata file for the service provider.
4. In the **Service Provider Name** field, type a unique name for the service provider.
5. If the metadata is signed, select the certificate from the **Select Signing Certificate** list.
6. Click **OK**.
   The file is uploaded, the SAML SP connector is created, and the window closes.

The SAML SP connector is displayed on the External SP Connectors list.

Binding a SAML IdP service to one SP connector

Bind a SAML Identity Provider (IdP) service and a SAML service provider (SP) connector so that the BIG-IP® system can provide authentication (SAML IdP service) to the external SAML service provider.

1. On the Main tab, click **Access** > **Federation** > **SAML Identity Provider**.
   The Local IdP Services screen opens.
2. Select a SAML IdP service from the list.

   Select an IdP service that you configured for use with one particular SP connector only.
3. Click **Bind/Unbind SP Connectors**.
   The screen displays a list of available SAML SP connectors.
4. Select the one SAML SP connector that you want to pair with this IdP service.
5. Select **OK**.
   The screen closes.

The SAML SP connector that you selected is bound to the SAML IdP service.

Configuring a full webtop

A full webtop allows your users to connect and disconnect from a network access connection, portal access resources, SAML resources, app tunnels, remote desktops, and administrator-defined links.

1. On the Main tab, click **Access** > **Webtops** > **Webtop Lists**.
   The Webtops screen displays.
2. Click **Create**.
   The New Webtop screen opens.
3. In the **Name** field, type a name for the webtop.
4. From the **Type** list, select **Full**.
   The Configuration area displays with additional settings configured at default values.
5. Click **Finished**.

The webtop is now configured, and appears in the list. You can edit the webtop further, or assign it to an access policy.

To use this webtop, it must be assigned to an access policy with an advanced resource assign action or with a webtop, links, and sections assign action. All resources assigned to the full webtop are displayed on the full webtop.

Configuring an access policy for a SAML SSO portal

Before you configure this access policy, configure an access profile without selecting an SSO configuration for it.

Configure an access policy so that the BIG-IP® system, as a SAML Identity Provider (IdP) can authenticate users using any non-SAML authentication type, and assign SAML resources and a webtop to the session.

***Note:** This access policy supports users that initiate a connection at a SAML service provider or at the SAML IdP.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Make any changes that you require to the logon page properties and click **Save**.
   The properties screen closes and the policy displays.
6. Add one or more authentication checks on the fallback branch after the **Logon Page** action.

   Select the authentication checks that are appropriate for application access at your site.
7. On a successful branch after an authentication check, assign SAML resources and a full webtop to the session.
   a) Click plus **[+]** on a successful branch after an authentication check.
      The Add Item window opens.
   b) On the Assignment tab, select the **Advanced Resource Assign** agent, and click **Add Item**.
      The Resource Assignment window opens.
   c) Click **Add new entry**.
      An **Empty** entry displays.
   d) Click the **Add/Delete** link below the entry.
      The screen changes to display resources on multiple tabs.
   e) Select the SAML tab, then from it select the SAML resources that represent the service providers that authorized users can access.
   f) Click **Update**.
      The window changes to display the Properties screen, where the selected SAML resources are displayed.
   g) Click the **Add/Delete** link below the entry.
      The screen changes to display resources on multiple tabs.

h) Select the Webtop tab, then select a full webtop on which to present the selected resources.

You must assign a full webtop to the session even if you have configured all SAML resources to not publish on a webtop.

i) Click **Update**.
The window changes to display the Properties screen. The selected webtop and SAML resources are displayed.

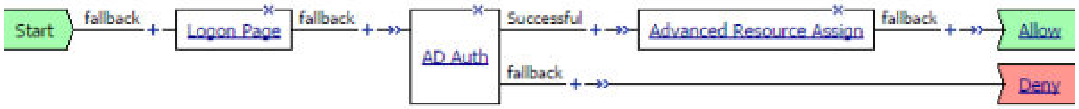
j) Click **Save**.
The Properties window closes and the Access Policy window is displayed.

You have configured a webtop to display resources that are available from service providers and that an authorized user can access.

8. (Optional) Add any other branches and actions that you need to complete the policy.
9. Change the Successful rule branch from **Deny** to **Allow**, and click the **Save** button.
10. Click the **Apply Access Policy** link to apply and activate the changes to the policy.
11. Click the **Close** button to close the visual policy editor.

You have an access policy that presents a logon page, authenticates the user, and assigns SAML resources and a full webtop on which to present them to the user.

Simple access policy for access to services on SAML service providers



To apply this access policy to network traffic, add the access profile to a virtual server.

***Note:** To ensure that logging is configured to meet your requirements, verify the log settings for the access profile.*

Verifying log settings for the access profile

Confirm that the correct log settings are selected for the access profile to ensure that events are logged as you intend.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. They enable and disable logging for access system and URL request filtering events. Log settings also specify log publishers that send log messages to specified destinations.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
The properties screen opens.
3. On the menu bar, click **Logs**.
The access profile log settings display.
4. Move log settings between the **Available** and **Selected** lists.

   You can assign up to three log settings that enable access system logging to an access profile. You can assign additional log settings to an access profile provided that they enable logging for URl request logging only.

   ***Note:** Logging is disabled when the **Selected** list is empty.*

5. Click **Update**.

An access profile is in effect when it is assigned to a virtual server.

Adding the access profile to the virtual server

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. Click **Update** to save the changes.

Your access policy is now associated with the virtual server.

OAuth Overview

APM OAuth 2.0 support

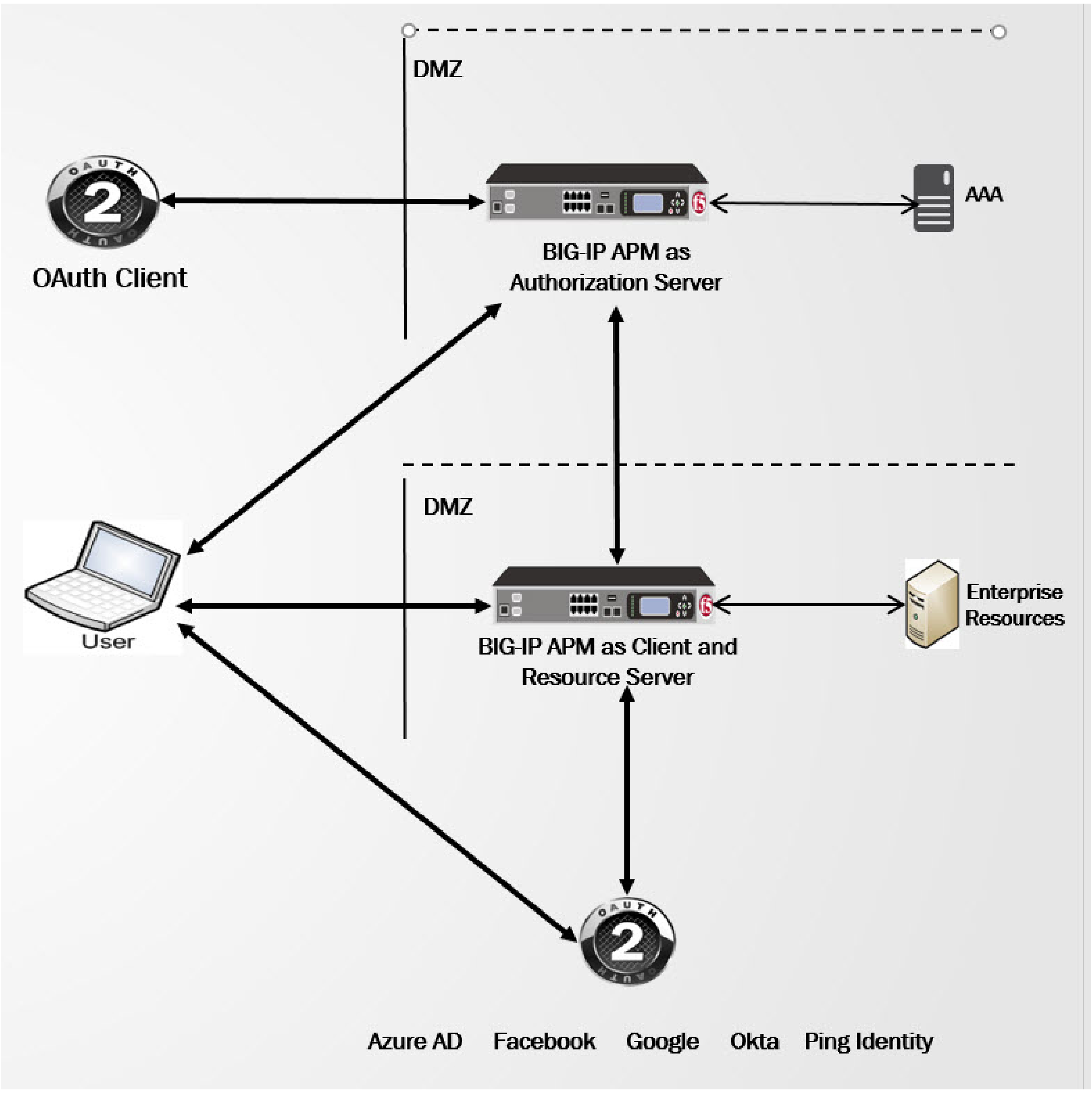


Figure 30: APM in OAuth roles in the network

OAuth roles that APM supports

On a single BIG-IP® system, Access Policy Manager® (APM®) can be configured to act as an OAuth 2.0 client and resource server, or to act as an OAuth 2.0 authorization server, or to act as both.

Important: *For APM to act in both OAuth roles on one BIG-IP system requires two virtual servers, one for each role, and two hostnames, one for each role.*

OAuth 2.0 roles

OAuth 2.0 specification RFC 6749 defines the roles in this table.

OAuth role	Description
resource owner	Can grant access to a protected resource. A resource owner can be an end-user (person) or another entity.
resource server	Hosts protected resources, and can accept and respond to requests for protected resources using access tokens.
client	Makes requests for protected resources on behalf of, and with authorization from, the resource owner. The client is an application.
authorization server	Issues access tokens to the client after successfully authenticating the resource owner and obtaining authorization.

APM in OAuth resource server and client roles

When Access Policy Manager® (APM®) acts as an OAuth resource server, users can log on using external OAuth accounts to gain access to the resources that APM protects. External OAuth accounts can be social accounts, such as Facebook and Google, or enterprise accounts, such as F5 (APM) and Ping Identity (PingFederate).

In this configuration, APM becomes a client application to an external OAuth authorization server, such as F5, on another BIG-IP® system, or Google.

APM in the OAuth authorization server role

When Access Policy Manager® (APM®) acts as an OAuth authorization server, APM can grant authorization codes, access tokens, and refresh tokens, and APM can perform token introspection.

Configuring one BIG-IP system for two OAuth roles

You can configure one BIG-IP® system with Access Policy Manager® (APM®) acting in an OAuth 2.0 client / resource server role and in an OAuth 2.0 authorization server role.

1. Configure APM to act as an OAuth client / resource server (or to act an OAuth resource server gateway).

   Follow the instructions in *BIG-IP® Access Policy Manager®: Authentication and Single Sign-On* on the AskF5™ web site located at `support.f5.com`.

   Note: *It doesn't matter whether you configure APM to act as an OAuth authorization server first or second. What's important is that you create separate virtual servers for each of the two configurations.*

2. Configure APM to act as an OAuth authorization server.

Follow the instructions in *BIG-IP® Access Policy Manager®: Authentication and Single Sign-On* on the AskF5™ web site located at `support.f5.com`.

Important: *In this step, be sure to configure a virtual server that is distinct from the one you configured for APM in the previous step.*

3. In your DNS configuration, configure two host names:

a) Point one hostname to the virtual server configured for APM as an OAuth authorization server.

b) Point the other hostname to the virtual server configured for APM in the other role (OAuth client / resource server or OAuth resource server gateway).

You have configured a BIG-IP system on which APM can play two OAuth roles.

Note: *Do not attempt to use a single FQDN to point to different ports on a single virtual server as an alternative; it does not work and is not supported.*

OAuth Client and Resource Server

OAuth 2.0 authorization servers that APM supports

Access Policy Manager® (APM®) supports OAuth 2.0 only. When configured as an OAuth client and resource server, APM has been tested with these OAuth authorization servers:

- AzureAD - Azure Active Directory
- F5 - APM configured as an OAuth authorization server.
- Facebook
- Google
- Okta
- Ping Identity - PingFederate

For compatibility information, see release notes for APM on the AskF5™ web site located at `support.f5.com`.

APM also supports the configuration of custom providers, that is, external OAuth 2.0 authorization servers that F5 has not tested. Custom providers should comply with RFC 6749.

About APM support for OpenID Connect

OpenID Connect adds an identity layer on top of OAuth 2.0. When configured as an OAuth client / resource server, Access Policy Manager® (APM®) can interact with an OpenID Connect provider to get this data:

UserInfo requests

APM can make UserInfo requests to an endpoint that is specified for that purpose on an OAuth provider. APM supports UserInfo requests from the OAuth Scope and OAuth Client agents in an access policy or a per-request policy subroutine.

ID Token

As defined in the OpenID Connect core 1.0 spec, an ID Token contains claims by an authorization server about the authenticated user when using a client. APM obtains an ID Token from an OAuth provider when OpenID Connect is enabled in the OAuth Client agent in an access policy or a per-request policy. (The OAuth provider must support OpenID Connect.)

Grant types that APM supports as an OAuth client

When configured as an OAuth client, Access Policy Manager® (APM®) supports authorization code and resource owner password credentials grant types.

About the OAuth client and resource server configuration

To configure Access Policy Manager® (APM®) as an OAuth client and resource server, first you must create these objects: OAuth providers, OAuth servers, and OAuth requests. Then, you must configure

APM policies with agents that reference the objects to get tokens, get permission for scopes, and retrieve scopes.

About OAuth providers

An OAuth provider configuration object specifies an external OAuth authorization server type and settings to support opaque access tokens, JSON web tokens, and ID Tokens and OpenID Connect UserInfo requests for those providers that support OpenID Connect.

About OAuth servers

An OAuth server specifies an OAuth provider and the OAuth role that Access Policy Manager® (APM®) plays with that provider. It also specifies the IDs, secrets, and SSL certificates that APM requires to communicate with the OAuth provider.

About OAuth requests

In Access Policy Manager® (APM®), the request object enables configuration of requests to meet the requirements of your OAuth providers. The object supports requests for scope permission, scope data, authorization redirect, tokens, and OpenID Connect UserInfo. It specifies the HTTP method, parameters, and headers to use for the specific type of request.

About OAuth Client

An OAuth Client agent is a policy item that requests authorization and tokens from an OAuth server. An OAuth Client can also get scope data on a per-request basis. The OAuth Client agent provides these configuration elements and options:

Server

Specifies the OAuth server to which this OAuth client directs requests.

Grant Type

Specifies the type of grant that the OAuth client uses.

- Authorization code - The client redirects the resource owner to the OAuth server to request an authorization code.
- Password - The client uses resource owner password credentials to request an access token from the OAuth server.

OpenID Connect

Specifies whether the agent uses OpenID Connect for authorization. Displays when **Grant Type** is set to **Authorization code**.

Note: *To function correctly when enabled, the OAuth provider (associated with the selected* ***Server****) must be configured to support JSON web tokens.*

OpenID Connect Flow Type

Specifies the OpenID Connect flow type to use: **Authorization code** or **Hybrid**.

OpenID Connect Hybrid Response Type

Specifies the response type to use for an OpenID Connect hybrid flow: **code-idtoken**, **code-token**, or **code-idtoken-token**.

Authentication Redirect Request

Specifies an auth-redirect-request type request, which redirects a user to an OAuth server. Displays when **Grant Type** is set to **Authorization code**.

Token Request
Specifies a token-request type of request.

Refresh Token Request
Specifies a token-refresh-request type of request. APM uses this request on a per-request basis.

OpenID Connect UserInfo Request
Specifies an openid-userinfo-request type of request. Displays when **OpenID Connect** is set to **Enabled**.

Redirection URI
Specifies the URI for the OAuth server to redirect a user back to the OAuth client. Displays when **Grant Type** is set to **Authorization code**.

Scope
Specifies one or more strings separated by spaces; for example `contacts photo email`. The strings are defined by the OAuth authorization server. Your best source of information for the strings that a particular OAuth authorization server defines could be APIs for OAuth 2.0 scopes on developer sites for OAuth providers.

For the **Authorization code** grant type, an OAuth authorization server prompts the user to grant or deny access to the scopes. For the **Password** grant type, an OAuth authorization server grants permission to the requested scopes based on the user providing resource owner password credentials.

Note: *Requests are configured in the* ***Access*** > ***Federation*** > ***OAuth Client / Resource Server*** > ***Requests*** *area of the product.*

About OAuth Scope

The OAuth Scope agent validates JSON web tokens (JWT) or validates scopes for opaque tokens. The OAuth Scope item provides these elements and options:

Token Validation Mode

- **Internal** - In this mode, the agent validates JSON web tokens (JWT).
- **External** - In this mode, the agent makes requests to an OAuth authorization server to get scopes associated with a token and to get scope data, such as a user's email address or contact list.

JWT Provider List
Specifies a list of OAuth providers that support JWT. The agent validates JWT from any of these providers when configured. For **Internal** mode.

Server
Specifies an OAuth server. OAuth servers in resource server, or client and resource server modes are available for selection. For **External** mode.

Scopes Request
Specifies a validation-scopes-request type request. This request type retrieves a list of scopes associated with the token. For **External** mode.

In **External** mode, there can be multiple scope data requests in this agent with these elements:

Scope Name
Specifies the name of a scope for which you are requesting data. (The external OAuth provider specifies the names of the scopes that it supports.)

Request

Specifies a scope-data-request type request. This is optional. If the provider does not require this type of request to obtain additional information from an authorization server, you do not need to fill in this field.

Note: *Requests are configured in the **Access** > **Federation** > **OAuth Client / Resource Server** > **Requests** area of the product.*

Overview: Configuring APM as an OAuth client and resource server

Register Access Policy Manager® (APM®) as a client to an external OAuth authorization server and get information from the external server about the scopes that it supports. Then configure the OAuth providers, servers, and requests to use from Access Policy Manager® (APM®) policies to interact with external OAuth authorization servers.

Task summary

Registering APM with a social media OAuth provider
Registering APM with an enterprise OAuth provider
Configuring OAuth providers to autodiscover JWTs and JWKs
Configuring JWKs for OAuth clients / resource servers
Specifying token configurations for JSON web tokens
Configuring OAuth providers without autodiscovery
Configuring OAuth servers for APM as client and resource server
Configuring OAuth servers for APM as a client
Configuring OAuth servers for APM as a resource server
Configuring requests for preconfigured providers
Configuring requests for custom providers
Configuring UserInfo requests for OpenID Connect
Configuring a provider list

Registering APM with a social media OAuth provider

For users to access resources that Access Policy Manager® (APM®) protects through social media accounts, you must add APM as a client application to an external server that provides OAuth authorization services.

Note: *Perform this step whether you plan to use APM in the OAuth client role, the OAuth resource server role, or in both the OAuth client and OAuth resource server roles.*

1. Create a developer account with the provider.

   Go to Google or Facebook developer sites or go to another OAuth provider site.
2. Log in to your developer account and add APM as a client application.

   Part of configuring APM as a client application should be to supply the provider with a redirect URI. The OAuth server uses it to redirect users back to APM OAuth client after they complete authorization. Type a URI that includes the FQDN for the virtual server you will configure on the BIG-IP system plus `/oauth/client/redirect`.

   Here is an example redirect URI: `https://siterequest.com/oauth/client/redirect`.
3. Copy the IDs and secrets from the configuration and keep them handy.

To configure APM as an OAuth client, you need to provide a client ID and client secret when you configure an OAuth server in APM.

To configure APM as an OAuth resource server, you need to provide a resource server ID and resource server secret when you configure an OAuth server in APM.

***Note:** Social media account providers supply only a client ID and client secret; you must use them for the client and resource server IDs and the client and resource server secrets in APM.*

Registering APM with an enterprise OAuth provider

To register Access Policy Manager® (APM®) as a client application of an enterprise provider, you must consult the enterprise provider documentation. For example, if F5 provides OAuth authorization services on another BIG-IP® system, you must register APM as a client or as a resource server on that BIG-IP system. F5 provides the necessary instructions in *BIG-IP® Access Policy Manager®: Authentication and Single Sign-On* on the AskF5™ web site located at `support.f5.com`.

For users to access resources that Access Policy Manager® (APM®) protects through enterprise accounts, you must add APM as a client application to an external server that provides OAuth authorization services.

***Note:** Perform this step whether you plan to use APM in the OAuth client role, the OAuth resource server role, or in both the OAuth client and OAuth resource server roles.*

1. Add APM as a client application of the enterprise OAuth authorization server following instructions from the enterprise provider.

   Part of configuring APM as a client application should be to supply the provider with a redirect URI. The OAuth server uses it to redirect users back to APM OAuth client after they complete authorization. Type a URI that includes the FQDN for the virtual server you will configure on the BIG-IP system plus `/oauth/client/redirect`.

   Here is an example redirect URI: `https://siterequest.com/oauth/client/redirect`.
2. Copy the IDs and secrets from the configuration and keep them handy.

   To configure APM as an OAuth client, you need to provide a client ID and client secret when you configure an OAuth server in APM.

   To configure APM as an OAuth resource server, you need to provide a resource server ID and resource server secret when you configure an OAuth server in APM.

Configuring OAuth providers to autodiscover JWTs and JWKs

You can automatically add JWT settings to Access Policy Manager® (APM®) if your OAuth provider supports OpenID Connect discovery.

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **Provider**.
   The Provider screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. From **Type**, select a type.

   You can configure autodiscovery for any provider type except Facebook.

   The fields that display and default values might change based on your choice.
5. Retain the selection of the **Use Auto-discovered JWT** check box.
6. In **OpenID URI**, type a string that appends `.well-known/openid-configuration` to the URL of the issuer and click **Discover**.
   Discovery updates the URIs in the remaining fields on the screen.

7. If any of the URIs contains a fragment, remove it before you save the provider.

   If there's a # character in a URI, you won't be able to save the provider.
8. Click **Save**.

Configuring JWKs for OAuth clients / resource servers

When Access Policy Manager® (APM®) is configured to act as an OAuth client or resource server, it uses JSON web keys (JWKs) to validate the JSON web tokens it receives. You can use APM to autodiscover JWKs from OAuth providers that support it. Otherwise, you can configure JWKs using this procedure.

1. On the Main tab, select **Access** > **Federation** > **JSON Web Token** > **Key Configuration**.
   The Key Configuration screen opens.
2. Click **Create**.
3. In the **Name** field, type a name.
4. In **ID**, type an ID.

   This parameter is used to identity a specific JSON web key.
5. For **Type**, select **RSA**, **Octet**, or **Elliptic Curve**.
   Additional parameters display for the type that you select.
6. (Optional) For **Signing Algorithm**, select any one.
7. For the **Octet** type, you only need to configure one additional setting:

   a) In **Shared Secret**, type a secret.

   ***Important:** To maximize the security of the algorithm, type enough characters so that the resulting key size matches the block size for the signing algorithm: for **HS256**, 32 characters; for **HS384**, 48 characters; for **HS512**, 64 characters.*

   b) Click **Save**.
   The newly created JWK displays on the list.
8. To support an **RSA** or an **Elliptic Curve** key type, you must either configure settings in the Certificates area or in the Parameters area. To get the necessary parameters from a certificate, go to the Certificates area and configure these settings:

   a) For **Certificate File**, select a certificate.

   ***Important:** When the BIG-IP® system is on a chassis platform or is included in an HA pair, do not select the default certificate. F5 strongly discourages the use of the default certificate in a JWK in any configuration.*

   ***Note:** The **Include X5C**, **Certificate Key**, and **Key Passphrase** settings apply only for a JWK for use with APM as an OAuth authorization server.*

   b) For **Certificate Chain**, select one.

   c) Click **Save**.
   The newly created JWK displays on the list.
9. For the **RSA** type, in the absence of a certificate go to the Parameters area and complete these substeps:

   a) For **Modulus**, type the modulus of the RSA public key.

   b) For **Public Exponent**, type the public exponent of the RSA public key.

   c) Supply values for the **SHA-1 Thumbprint**, and **SHA-256 Thumbprint** fields.

   d) Click **Save**.
   The newly created JWK displays on the list.
10. For the **Elliptic Curve** type, in the absence of a certificate, go to the Parameters area and complete these substeps:

a) For **X Coordinate**, type an X coordinate for the elliptic curve.
b) For **Y Coordinate**, type an Y coordinate for the elliptic curve.
c) For **Curve**, specify an elliptic curve.
   For example, type `P-256`.
d) (Optional) Supply values for the **SHA-1 Thumbprint**, and **SHA-256 Thumbprint** fields.
e) Click **Save**.
   The newly created JWK displays on the list.

Specifying token configurations for JSON web tokens

If you're configuring Access Policy Manager® (APM®) as an OAuth Client or OAuth Resource Server and your external OAuth provider supports OpenID Connect, you can autodiscover the token configuration from the provider instead of using this procedure to configure it. Before you use this procedure, configure JSON web key settings in APM.

You create token configurations for APM to use as an OAuth client/resource server.

1. On the Main tab, select **Access** > **Federation** > **JSON Web Token** > **Token Configuration**.
   The Token Configuration screen opens.
2. Open the properties for a token configuration:
   - To edit an existing token configuration, click its name.
   - To add a new token configuration, click **Create**, then on the screen that opens, type a name.
3. In **Issuer** field, type the URL of the token issuer, starting with `https`.

   Note: *You cannot modify the issuer if this JWT was auto-discovered.*
4. To use the value of **Access Token Expires In** from the provider list, retain the selection of the **Use Provider List Settings** check box. Otherwise, clear the check box and type the number of minutes the access token should live in the **Access Token Expires In** field.
5. For **Audience**, add all audiences that the provider supports. For each one, type a case-sensitive string and click **Add**

   Audience settings apply to the access token and not to the ID token.

   Interpretation of audience is application-specific.
6. In the Signing Algorithms area, select from the **Available** list and populate the **Allowed** and **Blocked** lists.
7. In the Keys (JWK) area, select from the **Available** list and populate the **Allowed** and **Blocked** lists.
8. In the Access Tokens area, for **Blacklist**, to reject a valid JWT access token that contains one of the configured claim names paired with one of the configured claim values, add names and values:
   a) In the Access Tokens area click (+).
   b) In the **Name** field, type a claim name.
   c) In the **Values** field, type a claim value to be blacklisted for that claim and click **Add**. (Repeat this step if necessary.)
9. Click **Save**.
   The new token displays on the Token Configuration list.

Configuring OAuth providers without autodiscovery

If you want to use JSON web tokens (JWTs) and the OAuth provider supports them, then configure JWTs in Access Policy Manager® (APM®) before you start this task.

You specify the URIs on an external server where APM gets OAuth authorization services when configured as an OAuth client and resource server.

Note: *APM provides preconfigured providers named* `AzureAD` *(Azure Active Directory from Microsoft),* `F5` *(APM),* `Facebook`*,* `Google`*,* `Okta`*, and* `Ping` *(PingFederate from Ping Identity).*

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **Provider**.
   The Provider screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. From the **Type** field, select a predefined type of provider, or select **Custom**.

   Based on the type that you select, the fields that display might change

   If you select a predefined type, additional fields fill with default values where possible.
5. If the **Use Auto-discovered JWT** setting displays, clear the check box.
   The **Token Configuration (JWT)** field displays and is set to **None**.
6. If these fields display and you want to support JWTs, configure them as follows:
   a) For **Token Configuration (JWT)**, select a configuration from the list.
   b) To specify that the OAuth provider ignore the certificate expiry check during token verification for a resource server, enable the **Ignore Expired Certificate Validation** check box.

   Important: *For this to work as intended, the OAuth authorization server must include an X5C (X. 509 Certificate Chain) parameter in its JWKS endpoint response.*
7. In the **Authentication URI** field, type the URI on the provider where APM should redirect the user for authentication.
8. In the **Token URI** field, type the URI on the provider where APM can get a token.
9. In the **Token Validation Scope URI** field, type the URI on the provider where APM can get information about a specific token.
10. If the provider supports OpenID Connect and you want to request identity information about a subject, for **UserInfo Request URI** type the URI at which to make the request.
11. Click **Finished**.
    The new provider displays on the Provider screen.

About OAuth client and resource server roles for APM

A **Mode** setting in the OAuth server configuration specifies the OAuth roles that you intend Access Policy Manager® (APM®) to play: OAuth client, OAuth resource server, or OAuth client and resource server. You can configure OAuth servers for each or any of the modes on one BIG-IP® system.

About SSL administration for OAuth clients and resource servers

You might need to perform some SSL administration tasks to support communication between Access Policy Manager® (APM®) as an OAuth client and an OAuth resource server, and external OAuth authorization servers. OAuth server objects on APM require a server SSL profile for an OAuth client and for an OAuth resource server.

SSL certificates are configured in the **System** > **Certificate Management** > **Traffic Certificate Management** area of the product.

Server SSL profiles are configured in the **Local Traffic** > **Profiles** > **SSL** > **Server** area of the product.

For more information, refer to *BIG-IP® System: SSL Administration* on the AskF5™ web site located at `support.f5.com`.

Configuring OAuth servers for APM as client and resource server

You configure the OAuth servers that process requests from Access Policy Manager® (APM®).

__Note:__ For APM to play the role of an OAuth client and an OAuth resource server, configure OAuth servers with __Mode__ set to __Client + Resource Server__.

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **OAuth Server**. The OAuth Server screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. From the **Mode** list, select **Client + Resource Server**.

   APM can use this OAuth server to request access tokens and scope details, such as an email address for the user.

   The Client Settings and Resource Server Settings areas display.
5. From the **Type** list, to get OAuth authorization services from another BIG-IP® system, retain the default selection **F5**; otherwise select another type.
   If any providers of the selected type exist, the **OAuth Provider** field displays one.
6. From the **OAuth Provider** list, retain the default selection or select another provider.
7. From the **DNS Resolver** list, select a DNS resolver (or click the plus (+) icon, create a DNS resolver, and then select it).
8. In the **Token Validation Interval** field, type a number.

   If you configure a per-request policy subroutine to validate the token, the subroutine repeats at this interval, or the expiry time of the access token, whichever is shorter.
9. In the Client Settings area, fill in these fields:

   You should have gotten a client ID and client secret when you registered APM as a client of the OAuth authorization server.

   a) In the **Client ID** field, type or paste the client ID.
   b) In the **Client Secret** field, type or paste the secret.
   c) From the **Client's ServerSSL Profile Name**, select a server SSL profile.
10. In the Resource Server Settings area, fill in these fields.

    You should have gotten an ID and secret from the OAuth authorization server when you registered APM with it.

    __Note:__ Social account providers supply only client ID and client secret. For social account providers, use the client ID and client secret for the client and the resource server IDs and secrets.

    a) In the **Resource Server ID** field, type or paste the resource server ID (for an enterprise provider).

       For a social provider, type or paste the client ID instead.
    b) In the **Resource Server Secret** field, type or paste the resource server secret (for an enterprise provider).

       For a social provider, type or paste the client secret instead.
    c) From the **Resource Server's ServerSSL Profile Name**, select a server SSL profile.
11. Click **Finished**.
    The server displays on the OAuth Servers screen.

The OAuth servers that you have configured are available for selection from the OAuth Client and OAuth Scope agents when you configure an access policy or a per-request policy.

Configuring OAuth servers for APM as a client

You configure the OAuth servers that process requests from Access Policy Manager® (APM®).

*Note: For APM to play the role of an OAuth client only, configure OAuth servers with **Mode** set to **Client**.*

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **OAuth Server**.
   The OAuth Server screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. From the **Mode** list, select **Client**.

   In this mode, APM can request access tokens from this OAuth server; APM can also refresh an existing access token when expired on a per-request basis.

   The Client Settings area displays.
5. From the **Type** list, to get OAuth authorization services from another BIG-IP® system, retain the default selection **F5**; otherwise select another type.
   If any providers of the selected type exist, the **OAuth Provider** field displays one.
6. From the **OAuth Provider** list, retain the default selection or select another provider.
7. From the **DNS Resolver** list, select a DNS resolver (or click the plus (+) icon, create a DNS resolver, and then select it).
8. If you have iRules® to use, in the **iRules** setting move them to the **Selected** list.

   For detailed information on iRules, see the F5 Networks DevCentral web site, `devcentral.f5.com`.
9. In the **Token Validation Interval** field, type a number.

   If you configure a per-request policy subroutine to validate the token, the subroutine repeats at this interval, or the expiry time of the access token, whichever is shorter.
10. In the Client Settings area, fill in these fields:

    You should have gotten a client ID and client secret when you registered APM as a client of the OAuth authorization server.

    a) In the **Client ID** field, type or paste the client ID.
    b) In the **Client Secret** field, type or paste the secret.
    c) From the **Client's ServerSSL Profile Name**, select a server SSL profile.
11. Click **Finished**.
    The server displays on the OAuth Servers screen.

The OAuth servers that you have configured are available for selection from the OAuth Client agent when you configure an access policy or a per-request policy.

Configuring OAuth servers for APM as a resource server

You configure the OAuth servers that process requests from Access Policy Manager® (APM®).

*Note: For APM to play the role of an OAuth resource server only, configure OAuth servers with **Mode** set to **Resource Server**.*

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **OAuth Server**.
   The OAuth Server screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.

4. From the **Mode** list, select **Resource Server**.

   APM can use this OAuth server to request a list of scopes associated with the access token , request details for a particular scope, such as an email address for the user, or to make both types of requests.

   The Resource Server Settings area displays.
5. From the **Type** list, to get OAuth authorization services from another BIG-IP® system, retain the default selection **F5**; otherwise select another type.
   If any providers of the selected type exist, the **OAuth Provider** field displays one.
6. From the **OAuth Provider** list, retain the default selection or select another provider.
7. From the **DNS Resolver** list, select a DNS resolver (or click the plus (+) icon, create a DNS resolver, and then select it).
8. If you have iRules® to use, in the **iRules** setting move them to the **Selected** list.

   For detailed information on iRules, see the F5 Networks DevCentral web site, `devcentral.f5.com`.
9. In the **Token Validation Interval** field, type a number.

   If you configure a per-request policy subroutine to validate the token, the subroutine repeats at this interval, or the expiry time of the access token, whichever is shorter.
10. In the Resource Server Settings area, fill in these fields.

    You should have gotten an ID and secret from the OAuth authorization server when you registered APM with it.

    Note: *Social account providers supply only client ID and client secret. For social account providers, use the client ID and client secret for the client and the resource server IDs and secrets.*

    a) In the **Resource Server ID** field, type or paste the resource server ID (for an enterprise provider).

       For a social provider, type or paste the client ID instead.

    b) In the **Resource Server Secret** field, type or paste the resource server secret (for an enterprise provider).

       For a social provider, type or paste the client secret instead.

    c) From the **Resource Server's ServerSSL Profile Name**, select a server SSL profile.
11. Click **Finished**.
    The server displays on the OAuth Servers screen.

The OAuth servers that you have configured are available for selection from the OAuth Scope agent when you configure an access policy or a per-request policy.

Configuring requests for preconfigured providers

Configure requests for token validation, getting scopes and scope details, authorization redirect, access token, or refresh token to meet the specifications of the provider.

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **Request**.
   APM supplies several preconfigured requests for these providers: AzureAD (Azure Active Directory), F5 (APM), Facebook, Google, Okta, and Ping (PingFederate from Ping Identity).
2. To copy a request:

   Note: *You cannot delete or update preconfigured requests. However, you can copy preconfigured requests and update the copies. You can also create new requests.*

   a) Click the **Copy** link.

   b) In the **New Request Name** field, type a name.

   c) Click **Copy**.
      The Properties screen for the new request displays.
3. For the **HTTP Method** and **Type** settings, retain the existing values or select new ones.

4. To edit existing request parameters:
   a) From **Request Parameters**, select an entry and click **Edit**.
   The entry populates the **Parameter Type** and **Parameter Name** fields, and, for a **custom** parameter type, the **Parameter Value** field.
   b) Update the newly populated fields.

   Important: *Do not configure a* `CODE`*,* `REFRESH_TOKEN`*, or* `STATE` *parameter. APM adds these parameters automatically when needed.*

   c) Click **Add**.
   The updated request parameters display in the list.
5. Add, delete, or edit any request parameters.
6. To add a request header:
   a) In the **Header Name** field, type a name.
   b) In the **Header Value** field, type a value.
   c) Click **Add**.

   The entry displays in the **Request Headers** field.
7. Click **Update**.

If using Okta preconfigured requests, Okta provides a refresh token only when `offline_access` is requested. Therefore, `offline_access` is used as the scope in default requests. If you need to make the authorize prompt visible, you must create a new request that does not specify `offline_access` as the scope.

Requests are available for selection in the OAuth Client and OAuth Scope agents when you configure an access policy or a per-request policy subroutine.

Configuring requests for custom providers

To perform this task, you need to understand the request parameters, request values, request headers, and HTTP method that the external OAuth authorization server supports for the type of request you plan to specify.

You configure a request for token validation, getting scopes and scope details, authorization redirect, access token, or refresh token that meets the specifications of the custom provider.

Note: *APM provides preconfigured providers, AzureAD (Azure Active Directory), F5 (APM), Facebook, Google, Okta, and Ping (PingFederate from Ping Identity), and supports configuration of custom providers.*

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **Requests**.
2. Click **Create**.
3. In the **Name** field, type a name.
4. For **HTTP Method**, retain **POST** or select **GET**.
5. From the **Type** list, select a request type.
6. If you selected the **scope-data-request** type, type a URI in the **URI** field.
7. Specify request parameters:
   a) From **Parameter Type**, select an option.
   If you select **custom**, the **Parameter Value** field displays.
   b) In the **Parameter Name** field, type a name as specified by your provider.

> ***Important:*** *Do not configure a* `CODE`, `REFRESH_TOKEN`, *or* `STATE` *parameter. APM adds these parameters automatically when needed.*

   c) In the **Parameter Value** field, type a value.
   d) Click **Add**.
      The entry displays in the **Request Parameters** field.

8. Add request headers if needed:
   a) In the **Header Name** field, type a name.
   b) In the **Header Value** field, type a value.
   c) Click **Add**.

   The entry displays in the **Request Headers** field.
9. Click **Finished**.

The request displays in the list on the screen.

Requests are available for selection in the OAuth Client and OAuth Scope agents when you configure an access policy or a per-request policy subroutine.

Configuring UserInfo requests for OpenID Connect

To perform this task, you need to understand the request parameters, request values, request headers, and HTTP method that the external OAuth authorization server supports for an OpenID Connect UserInfo request.

Access Policy Manager® (APM®) provides preconfigured requests for requesting UserInfo through OpenID Connect for the Ping (PingFederate from Ping Identity), Okta, and Google OAuth provider types. To get UserInfo from custom providers that support OpenID Connect, you configure requests that meet the specifications of the custom provider.

1. On the Main tab, click **Access** > **Federation** > **OAuth Client / Resource Server** > **Request**.
2. Click **Create**.
3. In the **Name** field, type a name.
4. For **HTTP Method**, retain **POST** or select **GET**.
5. From the **Type** list, select **openid-userinfo-request**.
6. Specify the options in the **Request Parameters** setting:
   a) From **Parameter Type**, select an option.
      If you select **custom**, the **Parameter Value** field displays.
   b) In the **Parameter Name** field, type a name as specified by your provider.

   > ***Important:*** *Do not configure a* `CODE`, `REFRESH_TOKEN`, *or* `STATE` *parameter. APM adds these parameters automatically when needed.*

   c) In the **Parameter Value** field, type a value.
   d) Click **Add**.
      The entry displays in the **Request Parameters** field.
7. Add **Request Headers** if needed:
   a) In the **Header Name** field, type a name.
   b) In the **Header Value** field, type a value.
   c) Click **Add**.

   The entry displays in the **Request Headers** field.
8. Click **Finished**.

Requests are available for selection in the OAuth Client and OAuth Scope agents when you configure an access policy or a per-request policy subroutine.

Configuring a provider list

Access Policy Manager® (APM®) provides a number of preconfigured OAuth providers. To add custom OAuth providers to a provider list, first configure them in the **Access** > **Federation** > **OAuth Client / Resource Server** > **Provider** area.

You configure a provider list for use with APM configured as an OAuth resource server. A *provider list* enables a single OAuth Scope agent in an access policy to validate tokens issued by multiple OAuth providers.

1. On the Main tab, select **Access** > **Federation** > **JSON Web Token** > **Provider List**.
2. Click **Create**.
3. In **Name**, type a name for the configuration.
4. In **Access Token Expires In**, type the number of minutes that access token should live.
5. From the **Provider** list, select a provider and click **Add**. Repeat this step to add additional providers to the list.

OAuth request type reference

The table lists OAuth request types, and specifies the agents that use them and the policies where they are used.

Request type	Description	Agents and policies where used
auth-redirect-request	Redirects a user to an authorization server.	An OAuth Client agent uses this request at the start of a session (from the access policy) and can also use it from a per-request policy subroutine. This request is applicable when the OAuth Client is configured with the authorization code grant. This request can only use the GET parameter.
token-request	Accesses an authorization server to obtain an access token or exchange an authorization code for an access token.	An OAuth Client agent uses this request at the start of a session (from the access policy) and can also use it from a per-request policy subroutine.
token-refresh-request	Refreshes an expired access token.	An OAuth Client agent can make token refresh requests from a per-request policy subroutine.
validation-scopes-request	Accesses an authorization server to get a list of scopes associated with an access token and to get scope data for those scopes.	An OAuth Scope agent uses this request at the start of a session (from the access policy) and can also use it from a per-request policy subroutine. The preconfigured **F5ScopesRequest** request is designed for use when APM is the OAuth server.
scope-data-request	Accesses an authorization server to get scope data.	An OAuth Scope agent can use this request type from an access policy and can also use it from a per-request policy subroutine.
openid-userinfo-request	Accesses a well-known endpoint for OpenID Connect to get UserInfo	An OAuth Scope and OAuth Client agent can use this request to get information.

Implementation result

The OAuth provider, server, and request objects are ready for use in OAuth Client and OAuth Scope agents in access policies and per-request policy subroutines.

Overview: Configuring policies for OAuth client and resource server

When Access Policy Manager® (APM®) acts as an OAuth client, an OAuth Client policy item can obtain an access token (and a refresh token if configured to do so) at the start of a session through the access policy. The OAuth client can also make OpenID Connect UserInfo requests following one of the OpenID Connect-defined flows (Authorization code flow or hybrid flow).

Throughout the session, an OAuth Client policy item can run periodically from a per-request policy subroutine to make OpenID Connect UserInfo requests, and, when the token expires, make an attempt to refresh the access token (if a refresh token exists) or authenticate the user anew.

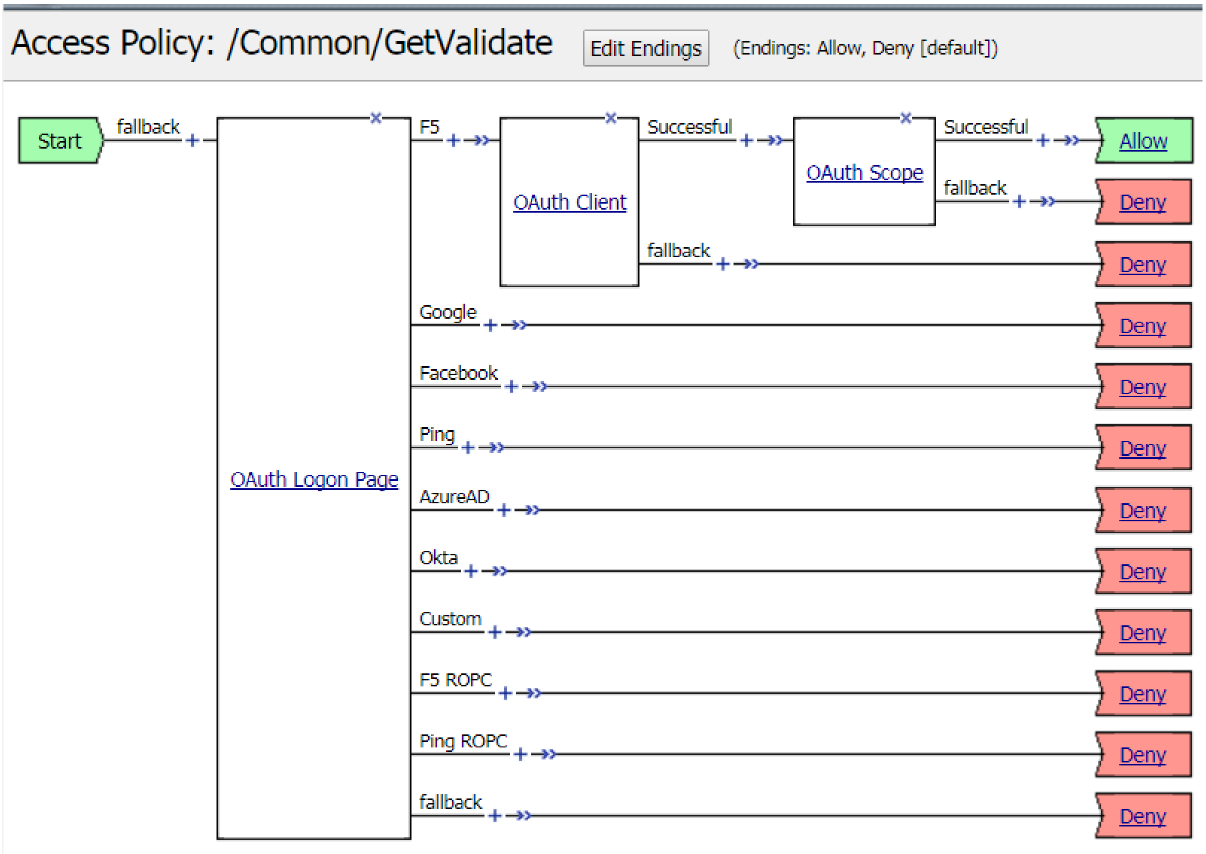
When APM acts as an OAuth resource server, an OAuth Scope policy item can be used to validate a token and make scope data and UserInfo requests from the access policy and, periodically, from a per-request policy subroutine.

Task summary

Creating an access profile for OAuth client and resource server
Configuring an OAuth Client agent in an access policy
Configuring OAuth Scope for opaque tokens in an access policy
Configuring OAuth Scope for JWTs in an access policy
Configuring an OAuth Client agent in a subroutine
Configuring OAuth Scope for opaque tokens in a subroutine
Configuring an OAuth Scope agent for JWTs in a subroutine
Creating a virtual server to manage HTTPS traffic
Adding access profile and per-request policy to the virtual server

Sample policy: Get an access token once per session

To obtain an access token one time only for the session, you need to configure an access policy. The OAuth Client agent can request a token.



With this access policy, the user logs on through a BIG-IP® system. The OAuth Client agent obtains an access token, and an OAuth Scope agent tries to retrieve the list of scopes associated with the token from the OAuth provider. The session starts. When the token expires, the system makes no attempt to refresh it.

Sample policies: Get a token and validate it for each request

To obtain an access token and to validate it for each request, first you need an access policy to obtain the token.

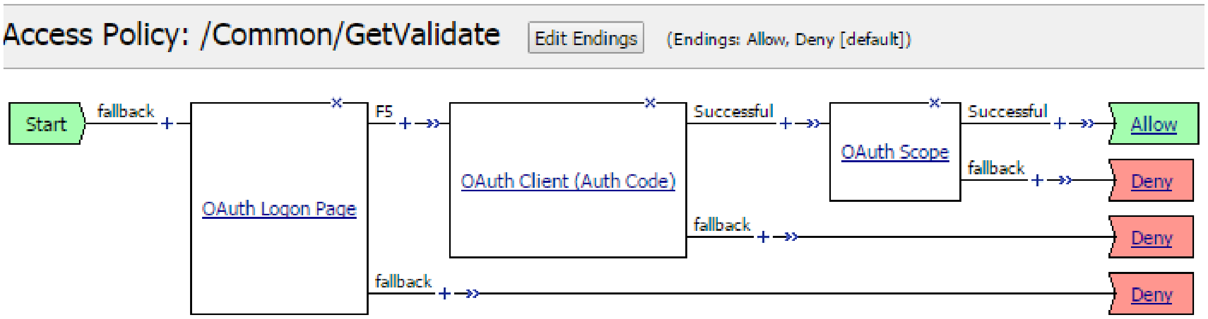
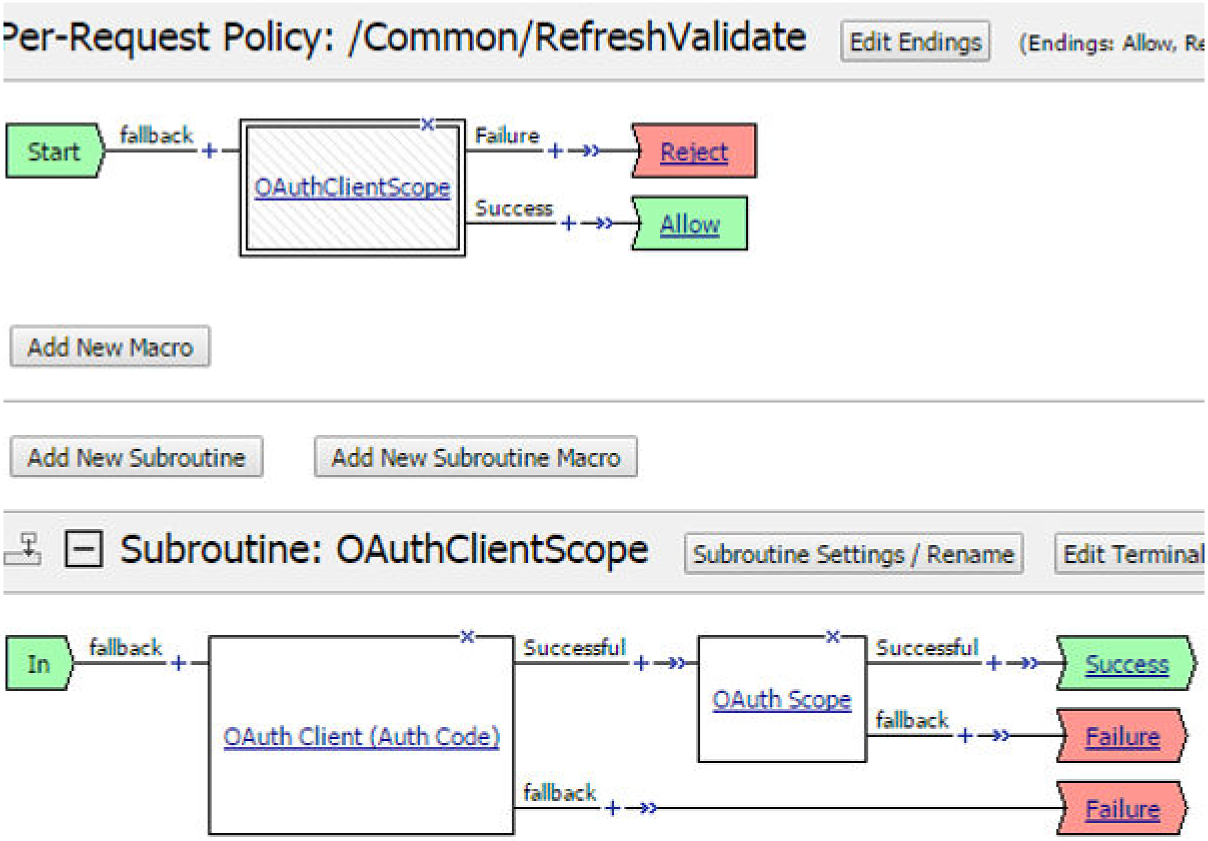


Figure 31: Access policy for APM as an OAuth client and resource server

To periodically validate and refresh the token, you need a per-request policy subroutine.

Figure 32: Per-request policy runs for each request but subroutine runs at an interval



Sample policy: Validate tokens per-request

When Access Policy Manager® (APM®) acts as an OAuth resource server, an OAuth Scope agent validates tokens obtained from the incoming request.

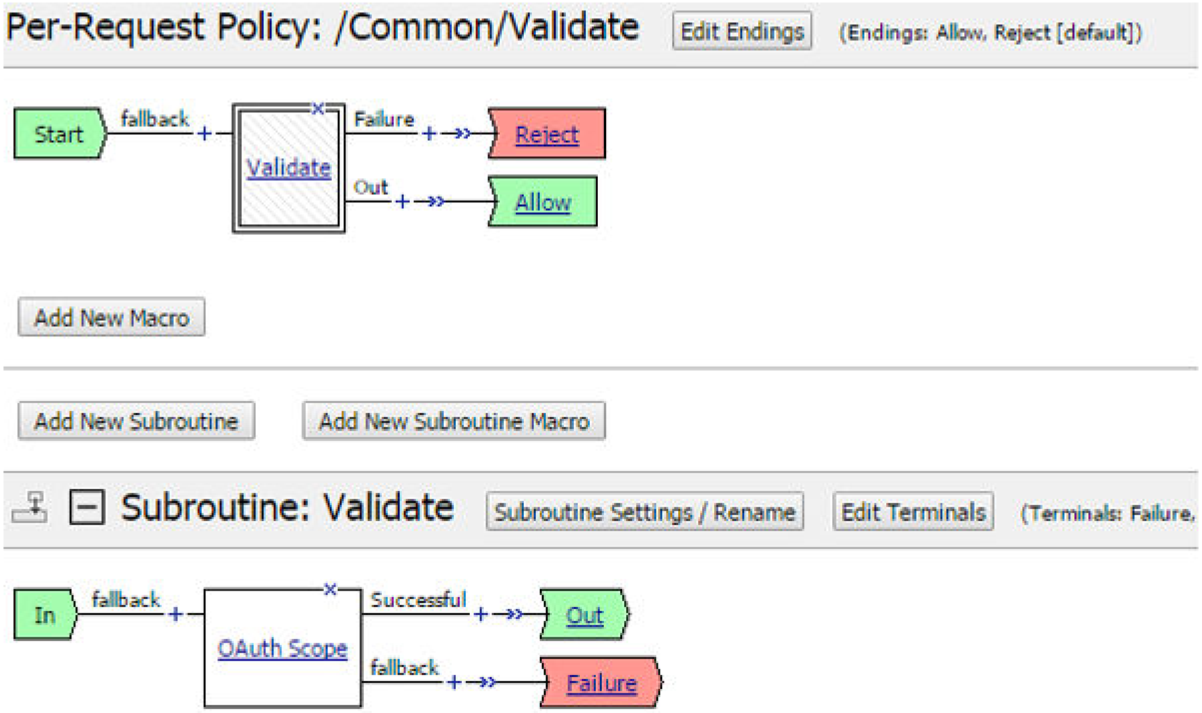


Figure 33: Per-request policy runs for each request, while subroutine runs only at an interval

Creating an access profile for OAuth client and resource server

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select one these options:
   - **LTM-APM**: Select for a web access management configuration.
   - **SSL-VPN**: Select to configure network access, portal access, or application access. (Most access policy items are available for this type.)
   - **ALL**: Select to support LTM-APM and SSL-VPN access types.
   - **SWG - Explicit**: Select to configure access using Secure Web Gateway explicit forward proxy.
   - **SWG - Transparent**: Select to configure access using Secure Web Gateway transparent forward proxy.

   Note: *Depending on licensing, you might not see all of these profile types.*

   Additional settings display.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.

6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Configuring an OAuth Client agent in an access policy

Before you start, you need an OAuth server configured in Access Policy Manager® (APM®) that supports clients (**Mode** is set to **Client** or **Client + Resource Server**) .

You add an OAuth Client agent to an access policy to request authorization from an OAuth server for one or more scopes. You can add more than one OAuth Client item to an access policy, with each item requesting permission for additional scopes.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the **Authentication** tab, select **OAuth Client**.
5. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
6. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
7. From the **Server** list, select an OAuth server.

   Only OAuth servers configured with **Mode** set to **Client** or **Client + Resource Server** display.
8. From the **Grant Type** list, select one of these options:

   - **Authorization code** - Redirects the user to the external server to authenticate. The user is redirected back to APM with an authorization code. APM uses the authorization code to request an access token
   - **Password** - Requests an access token from the external server by using the user's credentials (username and password). If this method is configured, the user must provide their external credentials to APM; to make this happen you must insert a logon page before the OAuth Client item in the access or the per-request policy.

   If you select **Authorization code**, the **Redirection URI** field displays.
9. To enable OpenID Connect in the agent, perform these substeps:
   a) For **OpenID Connect**, select **Enabled**.
   Additional fields display.
   b) For **OpenID Connect Flow Type**, retain **Authorization code** or select **Hybrid** and then select an entry for **OpenID Connect Hybrid Response Type**.
   c) For **OpenID Connect UserInfo Request**, select a request.
10. Select the requests to make to the OAuth server from the access policy.

- **Authentication Redirect Request** - Specifies an auth-redirect-request type request, which redirects a user to an OAuth server. Displays when **Grant Type** is set to **Authorization code**.
- **Token Request** - Specifies a token-request type of request.

You can make a selection from the **Refresh Token Request** list, but the OAuth Client does not use this request from an access policy.

Requests are configured in the **Access** > **Federation** > **OAuth Client / Resource Server** > **Requests** area of the product.

11. If the **Redirection URI** field displays, retain the default value (**https://%{session.server.network.name}/oauth/client/redirect**) or type a URI that points back to the APM client.

    Important: *If you type a URI, you must retain this path* `/oauth/client/redirect`*. Only change the host name portion of the URI.*

    The OAuth server uses the URI to send the user back to APM.
12. In the **Scope** field, type one or more scopes separated by spaces.

    Note: *Each time you add another OAuth Client agent to a policy, you must include the scopes (for example,* ***email photos****) that were requested in the previous instance of the OAuth Client and append any additional scopes (for example, contacts) to the list (for example,* ***email photos contacts****).*

    Read the OAuth provider documentation to learn the names of the scopes that they support and the URIs where you can obtain the data.
13. Click **Save**.
    The Properties screen closes. The newly added item displays in the policy.
14. If you selected **Password** from the **Grant Type** list, you must insert a logon page agent to precede the OAuth Client agent.
    a) Click (+) ahead of the **OAuth Client** on the policy branch.
    b) On the Logon Page tab, select **OAuth Logon Page** and click **Add Item**.
    A Properties screen displays.
    c) Click **Save**.
    The properties screen closes. The policy displays.
15. Complete the policy:
    a) Add any branch rules that you need.

    By default, the **OAuth Client** item has a successful branch for any valid non-error JSON response it receives. However, you can add other branch rules based on authorization server response to suit your needs.

    b) Change branch endings as needed; change **Deny** to **Allow** where you want to provide access.
16. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

Configuring OAuth Scope for opaque tokens in an access policy

Before you start, you need an OAuth server that supports resource servers (**Mode** is set to **Resource** or **Client + Resource Server**) configured in Access Policy Manager® (APM®).

You add an **OAuth Scope** item to a policy either to request the list of scopes associated with an opaque token or to request scope data from the OAuth server.

Note: *If the access policy already includes an* ***OAuth Client*** *item, place the* ***OAuth Scope*** *agent somewhere after the* ***OAuth Client*** *on the policy branch.*

---

1. On the Main tab, click **Access** > **Profiles / Policies**.

The Access Profiles (Per-Session Policies) screen opens.

2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the **Authentication** tab, select **OAuth Scope**.
5. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
6. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
7. From **Token Validation Mode**, retain the default setting, **External**.

   This mode allows the OAuth Scope agent to make scope requests and OpenID Connect UserInfo requests.
8. From the **Server** list, select an OAuth server.

   Only OAuth servers configured with **Mode** set to **Resource Server** or **Client + Resource Server** display.
9. To get a list of scopes associated with an access token, from the **Scopes Request** list, select a request to send to the OAuth provider.

   The list displays validation-scopes-request types.

   If F5 (APM) is the OAuth provider, select **F5ScopesRequest**.

   Requests are configured in the **Federation** > **OAuth Client / Resource Server** > **Requests** area of the product.
10. To add requests for scope data (for example, to request a user's email address or profile), perform these steps:
    a) Click **Add new entry**.
       A new line is added to the list of entries.
    b) In the **Scope Name** field, type the name of a scope that the OAuth provider supports.

       The scope must be associated with the access token. (The user must have granted permission for this scope.)

       For example, some OAuth providers support scopes named `email` or `profile`.
    c) From the **Request** list, select a request.

       The list includes scope-data-request types. Select one that you configured to meet the requirements of the specific OAuth provider.
11. To include a UserInfo request, select one from the **OpenID Connect UserInfo Request** list.
12. Click **Save**.
    The Properties screen closes. The newly added item displays in the policy.
13. Complete the policy:
    a) Add any additional policy items you require.
    b) Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.
14. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

Configuring OAuth Scope for JWTs in an access policy

Before you start, you need to have configured a JSON web token (JWT) provider list. You can configure it in the **Access** > **Federation** > **JSON Web Token** > **Provider List** area of the product.

You can add an **OAuth Scope** item to a policy to process JSON web tokens (JWTs) from multiple providers.

Note: *If the access policy already includes an* ***OAuth Client*** *item, place the* ***OAuth Scope*** *agent somewhere after the* ***OAuth Client*** *on the policy branch.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the **Authentication** tab, select **OAuth Scope**.
5. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
6. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
7. From **Token Validation Mode**, select **Internal**.

   This mode allows the OAuth Scope agent to process JSON web tokens from multiple providers.

   The **JWT Provider List** field displays.
8. From **JWT Provider List**, select a list.

   Provider lists are configured in the **Access** > **Federation** > **JSON Web Token** > **Provider List** area of the product.
9. Click **Save**.
   The Properties screen closes. The newly added item displays in the policy.
10. Complete the policy:
    a) Add any additional policy items you require.
    b) Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.

What causes a subroutine to run for an OAuth server?

A *per-request policy* runs at each request. A *per-request policy subroutine* runs periodically at whichever of these intervals is shorter: the number of minutes specified in the OAuth server **Token Validation Interval** field on Access Policy Manager® (APM®), or the interval that the external OAuth authentication server specifies in the `expires_in` attribute of the access token it issues.

Configuring an OAuth Client agent in a subroutine

Before you start, you need an OAuth server configured in Access Policy Manager® (APM®) that supports clients (**Mode** is set to **Client** or **Client + Resource Server**) .

You configure a per-request policy subroutine to periodically validate an OAuth token, to obtain a new or a refresh token, to get scopes and scope data.

1. On the Main tab, click **Access** > **Profiles / Policies** > **Per-Request Policies**.
   The Per-Request Policies screen opens.
2. In the **Name** field, locate the policy that you want to update, then in the **Per-Request Policy** field, click the **Edit** link.
   The visual policy editor opens in another tab.
3. Click the **Add New Subroutine** button.
   A popup screen opens.
4. On the **Subroutine from template** list, retain the selection **Empty**, and click **Save**.
   The popup screen closes. The subroutine, with the heading **[+] Subroutine:** ***`Name`***, displays below the main editor.
5. Expand the subroutine by clicking the [+] icon.
   The subroutine displays.
6. Click the **(+)** icon anywhere in the subroutine to add a new item.

   A small set of actions are provided for building a subroutine.

   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
7. On the **Authentication** tab, select **OAuth Client**.
8. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
9. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
10. From the **Server** list, select an OAuth server.

    Only OAuth servers configured with **Mode** set to **Client** or **Client + Resource Server** display.
11. From the **Grant Type** list, select one of these:

    - **Authorization code** - Redirects the user to the external server to authenticate. The user is redirected back to APM with an authorization code. APM uses the authorization code to request an access token
    - **Password** - Requests an access token from the external server by using the user's credentials (username and password). If this method is configured, the user must provide their external credentials to APM; to make this happen, you must insert a logon page before the OAuth Client item in the access or the per-request policy.

      Note: *If you select the password grant type, every time the per-request policy subroutine runs, it must request credentials from the user.*
12. To add an OpenID Connect UserInfo request to the policy, perform these steps:

    a) For **OpenID Connect**, select **Enabled**.
       Additional fields display.
    b) For **OpenID Connect Flow Type**, retain **Authorization code** or, select **Hybrid** and then select an entry for **OpenID Connect Hybrid Response Type**.
    c) For **OpenID Connect UserInfo Request**, select a request.

13. Select requests for the per-request policy to make to the OAuth server.

    - **Authentication Redirect Request** - Specifies an auth-redirect-request type request, which redirects a user to an OAuth server. Displays when **Grant Type** is set to **Authorization code**.
    - **Token Request** - Specifies a token-request type of request.
    - **Refresh Token Request** - Specifies a token-refresh-request type of request.

    Requests are configured in the **Access** > **Federation** > **OAuth Client / Resource Server** > **Requests** area of the product.
14. If the **Redirection URI** field displays, retain the default value (**https://%{session.server.network.name}/oauth/client/redirect**) or type a URI that points back to the APM client.

    ***Important:** If you type a URI, you must retain this path `/oauth/client/redirect`. Only change the host name portion of the URI.*

    The OAuth server uses the URI to send the user back to APM.
15. In the **Scope** field, type one or more scopes separated by spaces.

    ***Note:** Each time you add another OAuth Client agent to a policy, you must include the scopes (for example, **email photos**) that were requested in the previous instance of the OAuth Client and append any additional scopes (for example, contacts) to the list (for example, **email photos contacts**).*

    Read the OAuth provider documentation to learn the names of the scopes that they support and the URIs where you can obtain the data.
16. Click **Save**.
    The Properties screen closes. The newly added item displays in the policy.
17. If you selected **Password** from the **Grant Type** list, you must insert a logon page agent to precede the OAuth Client agent.
    a) Click (+) ahead of the **OAuth Client** on the policy branch.
    b) On the Logon Page tab, select **OAuth Logon Page** and click **Add Item**.
    A Properties screen displays.
    c) Click **Save**.
    The properties screen closes. The policy displays.
18. Complete the policy:
    a) Add any branch rules that you need.

    By default, the **OAuth Client** item has a successful branch for any valid non-error JSON response it receives. However, you can add other branch rules based on authorization server response to suit your needs.

    b) Change branch endings as needed; change **Deny** to **Allow** where you want to provide access.
19. Add the subroutine to the per-request policy:
    a) On a per-request policy branch, click the (+) icon.
    b) Select the Subroutines tab.
    c) Select the subroutine and click **Add Item**.
    The popup screen closes and the policy displays.
20. To rename the subroutine or to update number of seconds that the subroutine has to complete its interactions with the OAuth server, perform these steps:
    a) Click **Subroutine Settings/Rename**.
    b) To rename the subroutine, type in the **Name** field.
    c) To update the timeout, type a number in the **Subroutine Timeout (sec)** field.

Important: *No additional settings on this screen are applicable to the OAuth Client and OAuth Scope items.*

d) Click **Save**.

The popup screen closes. The subroutine displays in the policy.

To affect network traffic, along with an access policy, a per-request policy must be specified on a virtual server.

Configuring OAuth Scope for opaque tokens in a subroutine

Before you start, you need an OAuth server that supports resource servers (**Mode** is set to **Resource** or **Client + Resource Server**) configured in Access Policy Manager® (APM®).

To periodically get scopes and scope data for an existing access token, you configure an OAuth Scope agent in a per-request policy subroutine.

1. On the Main tab, click **Access** > **Profiles / Policies** > **Per-Request Policies**.
   The Per-Request Policies screen opens.
2. In the **Name** field, locate the policy that you want to update, then in the **Per-Request Policy** field, click the **Edit** link.
   The visual policy editor opens in another tab.
3. Click the **Add New Subroutine** button.
   A popup screen opens.
4. On the **Subroutine from template** list, retain the selection **Empty**, and click **Save**.
   The popup screen closes. The subroutine, with the heading **[+] Subroutine:** ***`Name`***, displays below the main editor.
5. Expand the subroutine by clicking the [+] icon.
   The subroutine displays.
6. Click the **(+)** icon anywhere in the subroutine to add a new item.

   A small set of actions are provided for building a subroutine.

   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
7. On the **Authentication** tab, select **OAuth Scope**.
8. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
9. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
10. From **Token Validation Mode**, retain the default setting, **External**.

    This mode allows the OAuth Scope agent to make scope requests and OpenID Connect UserInfo requests.
11. From the **Server** list, select an OAuth server.

    Only OAuth servers configured with **Mode** set to **Resource Server** or **Client + Resource Server** display.
12. To get a list of scopes associated with an access token, from the **Scopes Request** list, select a request to send to the OAuth provider.

    The list displays validation-scopes-request types.

    If F5 (APM) is the OAuth provider, select **F5ScopesRequest**.

    Requests are configured in the **Federation** > **OAuth Client / Resource Server** > **Requests** area of the product.

13. To add requests for scope data (for example, to request a user's email address or profile), perform these steps:
    a) Click **Add new entry**.
    A new line is added to the list of entries.
    b) In the **Scope Name** field, type the name of a scope that the OAuth provider supports.

       The scope must be associated with the access token. (The user must have granted permission for this scope.)

       For example, some OAuth providers support scopes named `email` or `profile`.
    c) From the **Request** list, select a request.

       The list includes scope-data-request types. Select one that you configured to meet the requirements of the specific OAuth provider.
14. To include a UserInfo request, select one from the **OpenID Connect UserInfo Request** list.
15. Click **Save**.
    The Properties screen closes. The newly added item displays in the policy.
16. Add the subroutine to the per-request policy:
    a) On a per-request policy branch, click the (+) icon.
    b) Select the Subroutines tab.
    c) Select the subroutine and click **Add Item**.
    The popup screen closes and the policy displays.
17. To rename the subroutine or to update number of seconds that the subroutine has to complete its interactions with the OAuth server, perform these steps:
    a) Click **Subroutine Settings/Rename**.
    b) To rename the subroutine, type in the **Name** field.
    c) To update the timeout, type a number in the **Subroutine Timeout (sec)** field.

       Important: *No additional settings on this screen are applicable to the OAuth Client and OAuth Scope items.*
    d) Click **Save**.

       The popup screen closes. The subroutine displays in the policy.

To affect network traffic, along with an access policy, a per-request policy must be specified on a virtual server.

Configuring an OAuth Scope agent for JWTs in a subroutine

Before you start this task, you need to have configured a JSON web token (JWT) provider list. Provider lists are configured in the **Access** > **Federation** > **JSON Web Token** > **Provider List** area of the product.

You can add an **OAuth Scope** item to a policy to process JSON web tokens (JWTs) from multiple providers.

1. On the Main tab, click **Access** > **Profiles / Policies** > **Per-Request Policies**.
   The Per-Request Policies screen opens.
2. In the **Name** field, locate the policy that you want to update, then in the **Per-Request Policy** field, click the **Edit** link.
   The visual policy editor opens in another tab.
3. Click the **Add New Subroutine** button.
   A popup screen opens.
4. On the **Subroutine from template** list, retain the selection **Empty**, and click **Save**.
   The popup screen closes. The subroutine, with the heading **[+] Subroutine: *`Name`***, displays below the main editor.

5. Expand the subroutine by clicking the [+] icon.
   The subroutine displays.
6. Click the **(+)** icon anywhere in the subroutine to add a new item.

   A small set of actions are provided for building a subroutine.

   A popup screen displays actions on tabs, such as General Purpose and Authentication, and provides a search field.
7. On the **Authentication** tab, select **OAuth Scope**.
8. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen opens.
9. In the **Name** field, type a name for the policy item.

   This name is displayed in the action field for the policy.
10. From **Token Validation Mode**, select **Internal**.

    This mode allows the OAuth Scope agent to process JSON web tokens from multiple providers.
11. From **JWT Provider List**, select a list.

    Provider lists are configured in the **Access** > **Federation** > **JSON Web Token** > **Provider List** area of the product.
12. Click **Save**.
    The Properties screen closes. The newly added item displays in the policy.

Creating a virtual server to manage HTTPS traffic

You can create a virtual server to manage HTTPS traffic.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Service Port** field, type `443` or select **HTTPS** from the list.
5. From the **HTTP Profile** list, select **http**.
6. For the **SSL Profile (Client)** setting, from the **Available** list, select **clientssl**, and using the Move button, move the name to the **Selected** list.
7. Click **Finished**.

The HTTPS virtual server appears in the Virtual Server List screen.

Adding access profile and per-request policy to the virtual server

You associate an access profile with a virtual server for use in establishing an access session. If you have configured a per-request policy, which is for use throughout the access session, you associate it with the virtual server also.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server that manages access for the web application you are securing.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. In the Access Policy area, from the **Per-Request Policy** list, select the policy that you configured earlier.
5. Click **Update**.

The access profile (and access policy) and the per-request policy are now associated with the virtual server.

Overview: Customizing an OAuth Logon Page

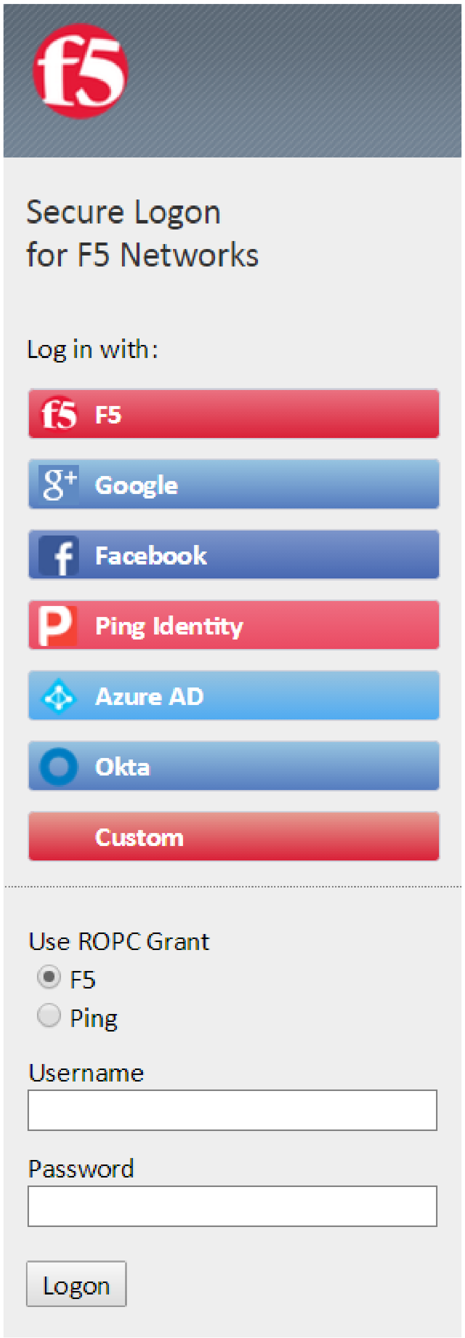
The default OAuth Logon Page agent displays messages that instruct the user to log in using an authorization code or a resource owner password credentials (ROPC) grant type. It displays option buttons from which users select the OAuth provider through whom they want to log in.

***Note:** For an alternative look and feel, an advanced customization template for the OAuth Logon Page agent is available on DevCentral.*

Customization options in the OAuth Logon Page agent enable an admin to add, update, delete, and reorganize the OAuth providers, as well as to configure the fields and text to display.

About the OAuth Logon Page advanced customization template

The OAuth Logon Page advanced customization template provides the code and the images necessary to display a logon page that, by default, looks like this.



Secure Logon
for F5 Networks

Log in with:

Use ROPC Grant

● F5

○ Ping

Username

Password

Figure 34: OAuth Logon Page (advanced customization template)

Note: *No affiliation between F5 Networks, Inc. (and its affiliated companies), and any of the above companies, including their associated products and services, relating to the functionality above, exists or is implied, nor is there any actual or implied recommendation or endorsement thereof.*

The template forms a starting point for additional customization to achieve a logon page with the preferred providers, images, colors, fields, and text.

The OAuth Logon Page template is available for download from DevCentral™ at `devcentral.f5.com`. Instructions for advanced customization with the OAuth Logon Page template are also available in *BIG-IP® Access Policy Manager®: Advanced Customization Examples* on DevCentral.

Updating OAuth providers on the OAuth Logon Page

You can add, delete, and reorder the list of OAuth providers that display on an OAuth Logon Page.

1. Open the per-session policy (or the per-request policy subroutine) that you want to update.
2. Click the **OAuth Logon Page** item.
   A Properties screen opens.
3. In row 1 of the Logon Page Agent table, click the **Values** field.

   Row 1 contains values for the `oauthprovidertype` variable.

   A popup screen displays options with a **Value** and a **Text** field for each provider.
4. Add, delete, or reorder the providers, and click **Finished**.
   The popup screen closes. The updated list of providers displays in the **Values** field in row 1.
5. If you added a provider, add a branch rule for that provider:
   a) Click the Branch Rules tab.
   b) Click **Add Branch Rule**.
      A new entry with **Name** and **Expression** settings displays.
   c) In the **Name** field, replace the default name by typing a new name.

      The name appears on the branch in the policy.
   d) Select and copy an expression from another branch rule.
      For example, copy the expression displayed for the **F5** branch rule:

```
expr {[mcget {session.logon.last.oauthprovidertype}] == "F5"}
```

   e) For the branch rule that you added, in the **Expression** setting click the **change** link.
      A popup screen opens.
   f) Click the Advanced tab, paste the expression into the field, and replace the existing provider name (such as `F5`) with the new provider name.
      For example, the result might look like this:

```
expr {[mcget {session.logon.last.oauthprovidertype}] == "Siterequest"}
```

   g) Click **Finished**.
      The popup screen closes.
   h) Click **Save**.
      The properties screen closes and the policy displays.
6. Click **Finished**.
   The popup screen closes.
7. If you are working with a per-session policy, click the **Apply Access Policy** link.

Overview: Configuring APM as an OAuth resource server gateway

Some applications do not support standard HTTP redirection and cookies. For Access Policy Manager® (APM®) to act as an OAuth resource server and provide an OAuth authorization layer into an API gateway, you must configure APM with a special access profile. This access profile also allows only a limited set of policy agents.

As a resource server gateway, APM must validate the tokens it receives. APM supports JSON web tokens (JWT) and validates them internally on the BIG-IP® system. APM also supports opaque tokens: to validate them, APM must interact with OAuth providers that are external to the BIG-IP system. APM also supports requesting UserInfo at a URI on an OAuth provider.

A single OAuth Scope agent works to validate tokens internally or externally. If you need to do both, you need two OAuth Scope agents. You can configure them using one access profile and policies that branch to the separate agents. Or you can configure them using two access profiles with policies that each contain an OAuth Scope agent. Those details are up to you. In this example, we walk through configuring one access profile and a policy that includes one OAuth Scope agent.

This configuration supports OAuth client apps that send requests with an HTTP Authorization header that contains an OAuth Bearer token.

Task summary

Creating an access profile for a resource server gateway
Creating APM access and per-request policies for a resource server gateway
Creating a virtual server to manage HTTPS traffic
Adding access profile and per-request policy to the virtual server

Creating an access profile for a resource server gateway

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select **OAuth-Resource Server**.
   The **User Identification Method** setting changes to **OAuth Token**.
5. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
6. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

Creating APM access and per-request policies for a resource server gateway

Before you start this task, you need to have configured these objects in Access Policy Manager® (APM®):

- An OAuth server that supports resource servers only (**Mode** is set to **Resource**).
- An access profile of the **OAuth-Resource Server** type.
- A JSON web token provider list to process JWTs if you plan to process them with this agent.

To use Access Policy Manager® (APM®) as an OAuth resource server gateway, you need an OAuth Scope agent to validate the tokens that APM receives. You can configure an OAuth Scope agent in an access policy, a per-request policy subroutine, or both.

Important: *In the policies, you must use only the subset of agents that are allowed with the* ***OAuth-Resource Server*** *access profile type.*

1. Open the access policy in the visual policy editor:
   a) On the Main tab, click **Access** > **Profiles / Policies** > **Access Profiles (Per-Session Policies)**.

The screen displays a list of access profiles.

b) Locate the correct access profile.

Look in the **Profile Type** field for **OAuth-Resource Server**.

c) In the **Per-Session Policy** field, click the **Edit** link.
The visual policy editor opens the access policy in a separate screen.

2. Click the **(+)** icon anywhere in the access policy to add a new item.

Note: *Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.

3. To review the subset of agents that are allowed for use with the **OAuth-Resource Server** access profile type, click each tab.
4. To configure the OAuth Scope agent to validate JSON web tokens internally, on the BIG-IP system, perform these substeps:

Note: *You can configure an OAuth Scope agent to validate JWTs internally or to perform validation externally. With external validation, you can validate opaque tokens and get UserInfo. To perform internal validation and external validation requires configuration of more than one OAuth Scope agent.*

To configure external validation instead, skip these substeps.

a) From **Token Validation Scope**, select **Internal**.

b) From **JWT Provider List** tab, select a list of providers.

c) Click **Save**.
The properties screen closes. The policy displays.

5. To configure an OAuth Scope agent to validate opaque tokens or to get UserInfo or both, perform these substeps:

a) From **Token Validation Scope**, select **External**.

b) On the **Authentication** tab, select **OAuth Scope**.

c) From the **Server** list, select an OAuth server that you know to be configured with **Mode** set to **Resource Server**.

Important: *Do not select an OAuth server configured with* ***Mode*** *set to* ***Client + Resource Server****.*

d) From the **Scopes Request** list, select a request.

Only validate-scopes-request type requests display.

If F5 (APM) is the OAuth provider, select **F5ScopesRequest**.

e) To add scope data requests, click **Add new entry** and, in the **Scope Name** field, type the name of a scope that the OAuth provider supports; then, from the **Request** list, select a scope data request that was configured to meet the requirements of the OAuth provider.

f) Click **Save**.
The properties screen closes. The policy displays.

6. To save and apply any changes that you made to the access policy, click **Apply Access Policy**.
7. To add an OAuth Scope agent to a per-request policy subroutine:

a) On the Main tab, click **Access** > **Profiles / Policies** > **Per-Request Policies**.

b) To open a per-request policy for editing, click the **Edit** link in the **Per-Request Policy** field.

Note: *If you must create a per-request policy first, provide a name for it that is unique among all access profiles and per-request policies.*

c) Click **Add New Subroutine**.
A popup screen opens. The **Subroutine from template** field specifies **Empty**.

d) Click **Save**.
The popup screen closes. The heading, **[+] Subroutine:** ***Name***, displays below the main editor.

e) Expand the subroutine for editing by clicking the **[+]** icon.

f) In the subroutine area, click **[+]**.
An **Add Item** popup screen opens.

g) On the Authentication tab, select **OAuth Scope** and click **Add Item**.

h) To configure the **OAuth Scope** agent properties, follow the instructions provided in this task previously.

i) To add the subroutine to the per-request policy, in the main editor click the **[+]** icon.

j) Click the Subroutines tab, select the subroutine, and click **Add Item**.
The subroutine displays in the per-request policy.

To put the access policy into effect, you must specify the access profile in the virtual server. If you configured a per-request policy subroutine, you must also specify the per-request policy in that virtual server.

Creating a virtual server to manage HTTPS traffic

You can create a virtual server to manage HTTPS traffic.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Service Port** field, type `443` or select **HTTPS** from the list.
5. From the **HTTP Profile** list, select **http**.
6. For the **SSL Profile (Client)** setting, from the **Available** list, select **clientssl**, and using the Move button, move the name to the **Selected** list.
7. Click **Finished**.

The HTTPS virtual server appears in the Virtual Server List screen.

Adding access profile and per-request policy to the virtual server

You associate an access profile with a virtual server for use in establishing an access session. If you have configured a per-request policy, which is for use throughout the access session, you associate it with the virtual server also.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server that manages access for the web application you are securing.
3. In the Access Policy area, from the **Access Profile** list, select the access profile that you configured earlier.
4. In the Access Policy area, from the **Per-Request Policy** list, select the policy that you configured earlier.
5. Click **Update**.

The access profile (and access policy) and the per-request policy are now associated with the virtual server.

Overview: Using an OAuth token for single sign-on

After Access Policy Manager® (APM®) gets or validates an OAuth token, the token can be used for single sign-on (SSO). Simply create an OAuth bearer SSO configuration and select it from any configuration object where APM lets you do that; for example, in an access profile.

APM gets or validates tokens when OAuth Client or OAuth Scope agents run in a policy.

Task summary

Creating an OAuth bearer SSO configuration

Adding OAuth bearer SSO to an access profile

Creating an OAuth bearer SSO configuration

You create an OAuth bearer SSO configuration when you want to allow single-sign on using an OAuth token that Access Policy Manager® (APM®) has gotten or validated from an external OAuth authorization server.

1. On the Main tab, select **Access** > **Single Sign-On** > **OAuth Bearer**.
   The OAuth Bearer Configurations screen opens.
2. Click **Create**.
   The New SSO Configuration screen opens.
3. In the **Name** field, type a name for the SSO configuration.
4. In the SSO Method Configuration area, select a server from the **OAuth Server** list.
5. Click **Finished**.

When you configure an APM object that supports single sign-on, it includes an **SSO Configuration** property. Select the SSO bearer configuration from the object where you want to put SSO into effect.

Adding OAuth bearer SSO to an access profile

You add an OAuth bearer SSO to an access profile if you want to allow SSO using an OAuth token from the access profile.

Note: *You can also select an SSO configuration from objects such as an Exchange profile, a SAML resource, or a portal access resource.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click the name of the access profile that you want to edit.
   The properties screen opens.
3. Click **SSO / Auth Domains** on the menu bar.
   The SSO Across Authentication Domains screen opens.
4. From the **SSO Configurations** list, select an OAuth bearer SSO configuration.

   Other settings on the screen are not relevant when using an OAuth bearer SSO configuration.
5. Click **Update**.

An access profile goes into effect when it is specified in a virtual server.

About OAuth statistics collection

Access Policy Manager® (APM®) collects OAuth performance statistics on the BIG-IP® system. After you configure and start to use APM as an OAuth server or an OAuth client and resource server, APM collects statistics without requiring any additional setup.

Charting OAuth client and resource server performance

To perform this task, you must be logged into the BIG-IP® system in one of these user roles: manager, administrator, or resource administrator.

You chart server performance when you want to see the rate of requests that Access Policy Manager® (APM®), acting as an OAuth client or resource server or both, processes over a period of time.

1. On the Main tab, click **Access** > **Overview** > **OAuth Reports** > **Client / Resource Server**.

   The x-axis for the chart specifies time. The y-axis specifies the rate: requests per second.

   The screen displays the Overview line graph with data from the last hour. The legend specifies the request types represented by each colored line; **Error** represents errors that occurred for any of the requests.
2. To display the rate of requests for the interval represented by a point in the chart, place your cursor at the point in the chart.
   The legend updates to display the rates.
3. To get the number of requests for a point in the chart, multiply the rate by the number of seconds in the interval.

   The interval between two consecutive points on a chart depends on the selected **Time**.
4. To generate the chart for another period of time, select one from the **Time** list.

OAuth performance chart intervals

The interval between two consecutive points in an OAuth performance chart depends on the time period selected for the chart.

Time	Interval
Last hour	30 seconds
Last 3 hours	1 minute (60 seconds)
Last 12 hours	6 minutes (360 seconds)
Last day	12 minutes (720 seconds)
Last week	1 hour (3600 seconds)
Last 30 days	4 hours (14400 seconds)
Last 3 months	12 hours (43200 seconds)
Last 6 months	1 day (86400 seconds)

OAuth client and resource server troubleshooting tips

You might run into problems with an OAuth client or resource server on the BIG-IP® system in some instances. Follow these tips to try to resolve any issues you might encounter.

Log message	Possible explanations and corrective actions
`OAuth Client: SSL profile from apmd is missing`	The server SSL profile is missing for the OAuth client or resource server that is specified in the OAuth server configuration. Verify the OAuth server configuration on the BIG-IP® system.
`Invalid URI log`	One or more URIs are not configured properly for the OAuth provider. Verify the OAuth provider configuration on the BIG-IP system.
`Failed to initialize OAuth agent`	This is an internal APMD error; this error should not occur.
`OAuth request is not configured for agent`	A particular request object is not configured properly. Verify request object configuration on the BIG-IP system.
`OAuth Client: state parameters do not match`	This is an internal APMD error; this error should not occur.
`OAuth Authorization redirect is empty`	The authorization redirect request URI is invalid. Verify the OAuth Client agent configuration in the policy and verify the configuration for the authorization redirect request that it specifies.
`Authorization code not found`	Authorization code sent by the authorization server is not found. There could be a problem on the OAuth client or on the external authorization server. Examine the error logs on the external authorization server.
`OAuth Client: Unsupported grant type`	An improper grant type was used from the OAuth Client; this error should not occur.
`OAuth Client: Unsupported parameter type`	A parameter for an OAuth request was invalid; this error should not occur.
`OAuth Response is empty from server`	The OAuth client (or resource server) received an empty response from the authorization server. Examine the error logs on the external authorization server.
`OAuth Scope: Failed to get scope details for scope`	The OAuth resource server failed to get scope details. Examine the error logs on the external authorization server.
`Unsupported agent type`	This is an internal APMD error; this error should not occur.
`OAuth Error Code received`	The OAuth client received an error code. Use the error code and message to help with troubleshooting.
`Access Token validation failed for server`	This warning means that the OAuth client agent failed to validate the token. When this happens, the OAuth client tries to refresh or get a new token
`OAuth Client: Failed to refresh token for server`	This warning means that the OAuth client failed to refresh an access token. When this happens, the OAuth client tries to fetch a new token.
`HTTP error 503, connect failed`	This error indicates that the OAuth client or resource server failed to connect to the authorization server. Typically, this is caused by one or

Log message	Possible explanations and corrective actions
	more configuration issues. Verify that these aspects of your configuration are correct: • DNS • Default gateway on the BIG-IP system • Proper SSL certificates on the OAuth server
`Invalid JWS token`	Configure JSON web keys (JWKs) to verify this JSON Web Token (JWT) signature; or, request a new valid JWT
`Invalid b64url encoded header`	The authorization server must include a base64 URL-encoded JSON Object Signing and Encryption (JOSE) header in the JWT.
`Invalid JSON`	Request a new valid JWT.
`Unsecured JWT should have alg none`	In an unsecured JWT, the authorization server must send a JOSE header with alg: none.
`Empty header: Algorithm(alg) field mandatory`	The authorization server must send a JOSE header with the "alg" claim and value.
`Empty payload: Issuer(iss) field mandatory`	The authorization server must send a JOSE payload with the "iss" claim and value.
`crit must be an array of string`	If the authorization server sends the "crit" header parameter, it must be a string array.
`crit must be a non-empty array of string`	If "crit" header parameter is specified in the JWT that the authorization server sends, it must be a non-empty string array.
`Invalid JOSE header parameter`	The JOSE header parameter sent in the JWT was invalid. Request a new valid JWT.
`Unsupported algorithm`	Request a JWT signed by an algorithm supported by F5 Client.
`Missing mandatory alg header parameter`	The authorization server must send a JOSE header with the "alg" claim and value.
`No provider supporting alg none is found with issuer`	Configure the provider that matches this issuer with allowed algorithms=none.
`Issuer Mismatch`	Configure the right issuer for this provider; or, request a JWT from the right provider.
`Audience not found`	The JWT was not meant for this client; or, configure the right audience for the client.
`JWT not active before nbf`	Invalid JWT received; request a new JWT.
`JWT expired`	Request another JWT, because the received JWT has expired.
`Audience is not string`	Configuration on the authorization server that sends JWT must send the audience claim with a string value.

Log message	Possible explanations and corrective actions
`Token blacklisted`	The received JWT has been blacklisted; request a new valid JWT.
`Signature verification failed`	Signature verification of the JWT did not succeed.
`Internal error during Signature verification`	Check JWK key configuration to match JWT signature
`Initialization failed`	Make sure the JWK key configuration matches the JWT algorithms
`JWT symmetric signature match failed`	Configure the correct symmetric key to verify this JWT signature.
`No JWK keys found to verify signature`	Configure the correct key to verify the received JWT signature.

OAuth Authorization Server

OAuth grant types

As an OAuth authorization server, Access Policy Manager® (APM®) supports the grant types in this table.

Grant Type	Description
Authorization code	An OAuth client directs a resource owner to an authorization server. As the OAuth authorization server, APM authenticates the resource owner and directs it back to the client with an authorization code. The client then uses the authorization code to get an access token.
Implicit	A client gets a token from the authorization server directly, based on resource owner authorization and without the exchange of intermediate credentials (such as an authorization code). This grant type is optimized for clients that are implemented using a scripting language in a browser. (Refresh tokens are not available with this grant type.)
Resource owner password credentials	A client goes directly to the authorization server and uses the resource owner credentials to obtain a token.

OAuth authorization server endpoints

As an OAuth authorization server, Access Policy Manager® (APM®) supports the endpoints listed in this table for interactions with resource owners and clients on the BIG-IP® system. APM supplies default URIs for each endpoint. Users can replace the default URIs.

Authorization Server Endpoint	Description
Authorization endpoint	As defined in the OAuth 2.0 authorization framework specification (RFC 6749), this endpoint is for use by a client to obtain authorization from the resource owner through user-agent redirection. The authorization server verifies the identity of the resource owner and interacts with the resource owner to obtain the authorization grant for the client. Defaults to `/f5-oauth2/v1/authorize`.
Token issuance endpoint	Specifies the endpoint for the client to use to obtain an access token or a refresh token, per RFC 6749. Defaults to `/f5-oauth2/v1/token`.
Token revocation endpoint	Specifies the endpoint for the client to use to revoke a previously obtained access token or

Authorization Server Endpoint	Description
	refresh token, as an extension of RFC 6749. Defaults to `/f5-oauth2/v1/revoke`.
Token introspection endpoint	As defined in the OAuth 2.0 token introspection specification (RFC 7662), clients and resource servers get information about the token, such as its status (active or not active), the scopes assigned to it, issue date, expiration date, and so on. Defaults to `/f5-oauth2/v1/introspect`.
OpenID Connect Configuration Endpoint	As defined in the OpenID Connect Discovery 1.0 specification, this defines the location of the OpenID provider configuration document. Defaults to `/f5-oauth2/v1/.well-known/openid-configuration`.

About OAuth token types

As an OAuth authorization server, Access Policy Manager® (APM®) supports bearer access tokens, and refresh tokens. For use as bearer access tokens and refresh tokens, APM supports opaque tokens and JSON web tokens.

About access tokens

As defined in the OAuth 2.0 specification (RFC 6749), an *access token* is a credential used to access protected resources. An access token is a string that represents an authorization issued to the client. A token represents specific scopes and durations of access granted by the resource owner. The resource server and the authorization server enforce the scopes and durations of access.

About refresh tokens

As defined in the OAuth 2.0 specification (RFC 6749), a *refresh token* is a credential used to obtain an access token. The client uses a refresh token to get a new access token from the authorization server when the current access token expires. If refresh tokens are enabled in the configuration, the OAuth authorization server issues a refresh token to the client when it issues an access token.

A refresh token is a string. It represents the authorization that the resource owner grants to the client. Unlike access tokens, a refresh token is for use with authorization servers only, and is never sent to a resource server.

About opaque tokens

Opaque tokens are issued in a proprietary format. Only the OAuth authorization server that issues the token can read it and validate it. The OAuth authorization server stores an opaque token for its lifetime and offers the ability to revoke the token. Use of opaque tokens forces client apps to communicate with the authorization server.

About JSON web tokens

JSON Web Token (JWT) is an open standard (*RFC 7519*) that defines a compact and self-contained way for securely transmitting information in a JSON object between OAuth entities. This information can be verified and trusted because it is digitally signed. JSON tokens are not stored on an OAuth authorization server and they cannot be revoked.

Overview: Configuring APM as an OAuth 2.0 authorization server

You can configure a BIG-IP® system with Access Policy Manager® (APM®) to act as an OAuth authorization server. OAuth client applications and resource servers can register to have APM authorize requests.

Task summary

Registering a client application for OAuth services
Registering a resource server for OAuth services
Configuring OAuth scopes of access for client apps
Configuring JWT claims for client apps
Configuring JWKs for OAuth authorization server
Managing storage for opaque tokens
Creating an OAuth profile
Creating an access profile for F5 as an OAuth authorization server
Configuring an access policy for F5 as an OAuth authorization server
Creating a client SSL profile for certificate inspection
Creating a virtual server for OAuth authorization server traffic

Registering a client application for OAuth services

For a client application to obtain OAuth tokens and OAuth authorization codes from the BIG-IP® system, you must register it with Access Policy Manager® (APM®).

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **Client Application**. The Client Application screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. In the **Application Name** field, type the application name.
5. In the Customization Settings for English area in the **Caption** field, type a caption.

   APM displays this caption as the name of the application on an Authorization screen if you choose to display one.
6. In the Security Settings area, for **Authentication Type**, select one of the options:
   - **None** - This is typically used in conjunction with the Implicit grant type, which does not use a secret or a certificate. For grant types other than **Implicit**, the other options provide better security.
   - **Secret** - This is the default setting. If this is selected, APM generates this secret for the client and you can request that APM regenerate the secret.
   - **Certificate** - Uses the client certificate. If this is selected, the **Client Certificate Distinguished Name** field displays.
7. If the **Client Certificate Distinguished Name** field displays, leave it blank or type a name.

   If you leave it blank, APM accepts any valid client certificate. If you specify a name, APM accepts only the specific valid client certificate with the specified Distinguished Name.

   This is a sample Distinguished Name for the client certificate:
   ```
   emailAddress=w.smith@f5.com,CN=OAuth AS Project Client2 Cert,OU=Product
   Development,O=F5 Networks,ST=CA,C=US
   ```
8. For **Scope**, select one or more and move them to the **Selected** field.

9. From **Grant Type**, select one or more of the options:
   - **Authorization Code** - The client must authenticate with the authorization server (APM) to get a token.
   - **Implicit** - The client gets a token from the authorization server (APM) without authenticating to it. (Refresh tokens are not available with this grant type.)
   - **Resource Owner Password Credentials** - The client goes directly to the authorization server and uses the resource owner credentials to obtain a token.
10. For **Redirect URI(s)** (if displayed), type a fully qualified URI, click **Add**, and repeat as needed.

    Redirect URI(s) form a list of URIs to which the OAuth authorization server can redirect the resource owner's user agent after authorization is obtained for an authorization code or implicit grant type.
11. To apply the token management settings from an OAuth profile, perform these substeps:
    a) In the Token Management Configuration area, retain selection of the **Enabled** check box.

       The token management configuration settings in an OAuth profile apply to client applications assigned to that profile except when this setting is disabled.

    b) Skip to step 13.
12. To manage tokens in a manner that is distinct for this client application, perform these substeps:
    a) In the Token Management Configuration area, clear the **Enabled** check box.
       Additional fields display.

    b) Update any of the additional fields.
13. Click **Finished**.

APM generates a client ID for the application. If the **Authentication Type** is set to **Secret**, APM generates a secret. The application displays on the Client Application screen.

Registering a resource server for OAuth services

For Access Policy Manager® (APM®) as an OAuth authorization server to accept token introspection requests from a resource server for token validation, you must register the resource server with APM.

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **Resource Server**.
   The Resource Server screen displays.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. For **Authentication Type**, select one of these:
   - **None** - This option requires no authentication when the resource server sends a token introspect request to the OAuth authorization server to get the token validated.
   - **Secret** - For this option, APM generates this secret and you can request that APM regenerate the secret.
   - **Certificate** - This is the default setting. If this is selected, **Resource Server Certificate Distinguished Name** field displays.
5. If **Resource Server Certificate Distinguished Name** displays, leave it blank or type a name.

   If you leave it blank, APM accepts any valid client certificate. If you specify a name, APM accepts only the specific valid client certificate with the specified Distinguished Name.

   This is a sample Distinguished Name for the client certificate:
   ```
   emailAddress=w.smith@f5.com,CN=OAuth AS Project Client2 Cert,OU=Product
   Development,O=F5 Networks,ST=CA,C=US
   ```
6. Click **Finished**.

The new resource server displays on the list.

Configuring OAuth scopes of access for client apps

When Access Policy Manager® (APM®) acts as an OAuth authorization server, you must configure scopes of access. (A *scope* specifies a string, and optionally, a value, that represents a resource.)

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **Scope**.
   The Scope screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. In the **Scope Name** field, type a name for the scope.
5. In the **Scope Value** field, type a value for the scope.
6. In the Customization Settings for English area, in the **Caption** field, type a caption.
7. In the **Detailed Description** field, type a description of the access that the client application needs.

   If you choose to display an Authorization screen, APM displays the contents of this field on it; or, if this field is blank, APM displays the contents of the **Caption** field.

   Here are some examples: `Access your profile` or `Update your tasks, projects, and workspace`.
8. Click **Finished**.

Configuring JWT claims for client apps

When Access Policy Manager® (APM®) acts as an OAuth authorization server, you must configure the claims that you want APM to include in the JSON web tokens it generates. (A *claim* specifies a string, and optionally, a value, that represents a resource.)

1. On the Main tab, select **Access** > **Federation** > **OAuth Authorization Server** > **Claim**.
2. Click **Create**.
3. In **Name**, type a name for the configuration.
4. In **Claim Name**, type a name for the claim.
5. In **Claim Value**, type a value for the claim.
6. Click **Save**.
   The newly created claim displays on the list.

You associate claims with tokens when you configure an OAuth profile or a client application.

Configuring JWKs for OAuth authorization server

You configure JSON web keys (JWKs) for Access Policy Manager® (APM®) to use to sign the JSON web tokens that it issues when APM acts as an OAuth authorization server.

1. On the Main tab, select **Access** > **Federation** > **JSON Web Token** > **Key Configuration**.
   The Key Configuration screen opens.
2. Click **Create**.
3. In the **Name** field, type a name.
4. In **ID**, type the ID.
5. For **Type**, select **RSA**, **Octet**, or **Elliptic Curve**.
   Additional parameters display for the type that you select.
6. For **Signing Algorithm**, select any one.
7. For the **Octet** type, you only need to configure one additional setting:

a) In **Shared Secret**, type the secret.

Important: *To maximize the security of the algorithm, type enough characters so that the resulting key size matches the block size for the signing algorithm: for* ***HS256****, 32 characters; for* ***HS384****, 48 characters; for* ***HS512****, 64 characters.*

b) Click **Save**.
The newly created JWK displays on the list.

8. For the **RSA** or **Elliptic Curve** key types, configure the settings in the Certificate areas:

a) For **Certificate File**, select a certificate.

Important: *Do not select the default certificate when the BIG-IP® system is on a chassis platform or is included in an HA pair. F5 strongly discourages the use of the default certificate in a JWK in any configuration.*

b) To include an X5C (X.509 Certificate Chain) parameter in the JWKS response from the OAuth authorization server JWKS endpoint, select **Include X5C**.

c) For **Certificate Key**, select one.

Important: *Do not select the default key when the BIG-IP system is on a chassis platform or is included in an HA pair. F5 strongly discourages the use of the default key in a JWK in any configuration.*

d) For **Key Passphrase**, type a passphrase.

e) For **Certificate Chain**, select one.

9. Click **Save**.
The newly created JWK displays on the list.

Managing storage for opaque tokens

You create database instances to store the opaque tokens that Access Policy Manager® (APM®) grants and then stores for the tokens' lifetimes.

Note: *APM provides one default database instance,* `oauthdb`*. Additional instances enable you to group tokens.*

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **Database Instance**.
   The Database Instance screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. In the Purge Schedule Settings area, select a frequency from the **Frequency** list and specify a time in the **Schedule At** field.

   Schedule the database purge for a time when the BIG-IP® system is least used to prevent any possible performance issues.

   Purging removes expired access tokens, refresh tokens, authorization codes, and associated entries from the instance. For the purpose of purging, an access token is considered expired when it passes the date when it expires; (expiry is based on the **Access Token Lifetime** setting).

   Note: *Expired access tokens are not removed when the* ***Reuse Access Token*** *setting is enabled (in the corresponding OAuth profile) and a refresh token has been issued and the refresh token is not expired.*

   Revoked access tokens are purged after they expire.
5. To save this database instance, click **Finished**.

Database instances are available for selection in an OAuth profile.

Creating an OAuth profile

You configure an OAuth profile to specify the client applications, resource servers, token types, and authorization server endpoints that apply to the traffic that goes through a particular virtual server.

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **OAuth Profile**.
   The OAuth Profile screen opens.
2. Click **Create**.
3. In the **Name** field, type a name for the object.
4. For **Client Application**, select from the available clients and move them to the **Selected** list.
5. For **Resource Server**, select from the available servers and move them to the **Selected** list.
6. Click **Finished**.

You have created an OAuth profile that supports the client apps and resource servers you selected; it supports opaque tokens and is configured to store them in the default database instance.

You can update the types of tokens (JSON web token and opaque token) provided through this OAuth profile and update token management settings for either type of token.

Enabling or disabling opaque tokens and JSON web tokens

Before you begin this task, you must create an OAuth profile.

You configure the OAuth profile so that the OAuth authorization server can issue opaque tokens, JSON web tokens (JWT), or both, for the traffic that goes through a particular virtual server.

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **OAuth Profile**.
   The OAuth Profile screen opens.
2. Click the name of the OAuth profile you want to edit.
3. In the Token Management Configuration area, select the **Custom** check box.
   Settings become available.
4. To update support for opaque tokens, locate the **Support Opaque Token** check box; then select it to enable opaque tokens or clear it to disable them.
   When the check box is selected, settings for opaque tokens display, and when it is cleared the settings are hidden.
5. To update support for JSON web tokens, locate the **Support JWT Token** check box; then select it to enable JWT tokens or clear it to disable them.
   When the check box is selected, settings for JWT tokens display, and when it is cleared the settings are hidden.
6. Click **Update**.

If the OAuth profile supports both opaque tokens and JWTs, for an OAuth client to get a JWT, its request to the authorization server must include this parameter and value: `token_content_type=jwt`.

Configuring opaque token settings in an OAuth profile

Before you start, configure an OAuth profile. By default, an OAuth profile enables opaque tokens and supplies default token management settings for them.

You might want to store opaque tokens in a non-default database instance or change the access token lifetime.

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **OAuth Profile**.
   The OAuth Profile screen opens.
2. Click the name of the OAuth profile you want to edit.

3. In the Token Management Configuration area, select the **Custom** check box.
   Settings become available.
4. From the **Database Instance** list, you can retain the default, **oauthdb**, or select another database instance.
5. To update endpoints:
   a) In the Authorization Server Endpoints area, select the **Custom** check box.
      Settings become available.
   b) Change values in any of these fields: **Authorization Endpoint**, **Token Issuance Endpoint**, **Token Revocation Endpoint**, and **Token Introspection Endpoint**.
6. Click **Update**.

Configuring support for JWTs in an OAuth profile

Before you start, configure an OAuth profile, configure JSON web keys (JWK), and configure claims.

***Note:** You can configure JWKs in the **Access** > **Federation** > **JSON Web Token** area of the product.*

So that Access Policy Manager® (APM®) will generate JSON web tokens (JWTs) for the traffic on a specific virtual server, you configure these settings in the OAuth profile.

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **OAuth Profile**.
   The OAuth Profile screen opens.
2. Click the name of the OAuth profile you want to edit.
3. In the Token Management Configuration area, select the **Custom** check box.
   Settings become available.
4. If the **Support JWT Tokens** check box is cleared, select it.
   Additional settings display.
5. In **Issuer**, type the URL for the issuer.
   For example, type `https://big-ip-server.com` where *`big-ip-server`* is the name of your server.
6. In **Subject**, retain the default value, `%{session.assigned.uuid}`, or type a subject for the JWT.

   The session variable `session.assigned.uuid` contains the UUID that APM assigns to the session after the access policy completes.
7. For **Primary Key**, select a JWK from the list.

   ***Important:** Key rotation is a manual process. The administrator should keep track of the certificate expiration for the primary key and assign rotation keys as needed.*
8. To specify **Rotation Keys**, select one or more JWKs and move them to the **Selected** list.
9. To specify audience claims, in the **Audience** field, type a string and click **Add**. Repeat this step as needed.
10. To specify claims, for **Claims** move claims to the **Selected** list.
11. In **JWT Refresh Token Encryption Secret**, type a string.

    ***Important:** If the **JWT Generate Refresh Token** setting is enabled, after you set this secret do not change it. Changing the secret automatically invalidates all the issued refresh tokens.*

    ***Note:** F5 recommends that you write the secret down and store it in a safe place in case you ever need to rebuild the OAuth profile.*
12. To update endpoints, in the Authorization Server Endpoints area select the **Custom** check box.
    Settings become available.

13. To update the OpenID Connect discovery endpoint, in **OpenID Connect Configuration Endpoint** type the URI where clients can find the OpenID Connect provider configuration document.
14. To update the JSON Web Key Set endpoint, in **JWKS Endpoint**, type the URI where clients can locate the public signing keys for the APM OAuth authorization server.
15. Click **Update**.

If the OAuth profile supports both opaque tokens and JSON web tokens (JWT), for an OAuth client to get a JWT, its request to the authorization server must include this parameter and value: `token_content_type=jwt`.

About key rotation for JWTs

Access Policy Manager® (APM®) does not support automatic rotation of signing keys for JSON web tokens (JWTs). To configure signing keys, an administrator selects a primary key in the OAuth profile for authorization server configurations, and optionally, can specify rotation keys. To determine when to update the primary key and when to add or to update rotation keys, an administrator might consider factors such as when the certificates in the keys expire, and how long JWTs that use a particular key remain valid.

Creating an access profile for F5 as an OAuth authorization server

You create an access profile to provide the access policy configuration for a virtual server that establishes a secured session. Configure an access profile like this for traffic to Access Policy Manager® (APM®) as an OAuth authorization server.

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.

   Note: *A access profile name must be unique among all access profile and any per-request policy names.*

4. From the **Profile Type** list, select **All** or **LTM-APM**.
5. Scroll down to the Configurations area.
6. From the **OAuth Profile** list, select the OAuth profile you configured earlier.
7. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.

   Important: *If you want to translate text into other languages (as you can in Access Policy Customization), make sure to select the languages that you want to display here.*

8. Click **Finished**.

The access profile displays in the Access Profiles List. Default-log-setting is assigned to the access profile.

When you create a virtual server to process traffic from OAuth version 2.0 clients and resource servers, assign this access profile to it.

Sample policy: Logon, authenticate, and authorize

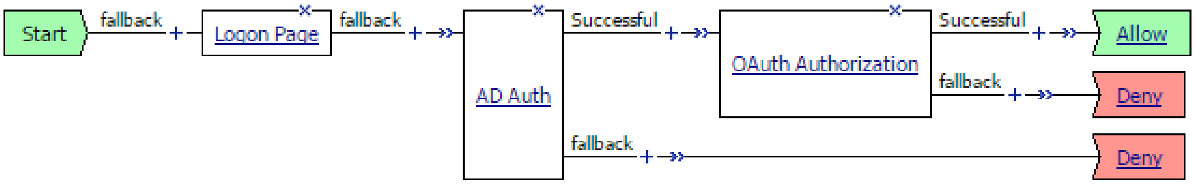


Figure 35: Access policy for APM as an OAuth authorization server

The Logon Page and OAuth Authorization agents are required in the access policy for Access Policy Manager® (APM®) to act as an OAuth authorization server. An authentication agent, such as AD Auth, is optional; if included in a policy, an authentication agent should be placed after the Logon Page and before the OAuth Authorization agent.

About OAuth Authorization

When Access Policy Manager® (APM®) is configured to act as an OAuth authorization server, an OAuth Authorization agent must be present in the access policy.

The OAuth Authorization agent provides these elements and options.

Prompt for Authorization

- **Enabled** - Displays the OAuth Authorization page. The page requests authorization for the client application to access a list of scopes and presents the options to allow or to deny access.
- **Disabled** - Does not display the OAuth Authorization page.

Subject

Type the name of a subject claim (for JSON web tokens).

Audience

Specifies the audiences for the claims (for JSON web tokens).

Scope / Claim Assign

Specifies the scopes or the claims for which authorization is requested. If no scopes or claims are specified here, the ones configured in APM for the client application are used.

Customization

Customize the messages that display on the OAuth authorization page when **Prompt for Authorization** is set to **Enabled**:

- **Authorize Message** Specifies the initial wording for the prompt.
- **Scope Message** Specifies the wording that precedes the list of scopes that are specified in the Scope / Claim Assign area of this screen.
- **Allow Message** Provides the label for the button that allows access.
- **Deny Message** Provides the label for the button that denies access.

Configuring an access policy for F5 as an OAuth authorization server

You configure an access policy so that, as OAuth authorization server, Access Policy Manager® (APM®) can identify and authorize client applications to access resources.

***Note:** The policy items in these steps are necessary to process traffic sent to F5 (APM) as an authorization server. You can add these items to a branch of an existing policy or add them to a new policy.*

1. On the Main tab, click **Access** > **Profiles / Policies**.
   The Access Profiles (Per-Session Policies) screen opens.
2. In the Per-Session Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** icon anywhere in the access policy to add a new item.

   ***Note:** Only an applicable subset of access policy items is available for selection in the visual policy editor for any access profile type.*

   A popup screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. On the Logon tab, select **Logon Page** and click the **Add Item** button.
   The Logon Page Agent properties screen opens.
5. Click **Save**.
   The properties screen closes and the policy displays.
6. On a policy branch, click the **(+)** icon to add an item to the policy.
7. On the Authentication tab, select **OAuth Authorization** and click **Add Item**.

   ***Important:** You must include an **OAuth Authorization** item in the policy for it to work.*

   A Properties popup screen opens.
8. If you do not want to prompt for authorization, in the OAuth Authorization area, from the **Prompt for Authorization** list, select **Disabled**.
9. In the **Subject** field, type the subject.

   This is the subject of a JSON web token (JWT).
10. In the Audience area, for each audience that you want to support for JWT:
    a) Click **Add new entry**.
       A numbered entry displays.
    b) Type an audience name in the new field.
11. In the Scope / Claim Assign area, add entries to assign scopes, claims, or both:

    Assign these whether or not you plan to prompt for authorization.

    a) Click **Add new entry**.
       A numbered entry displays with **Expression** and **Claim** and **Scope** properties.
    b) To specify a prerequisite for the scopes and claims, click **change** and configure an expression.

       A prerequisite is not mandatory.

       For example, use an expression to verify that the user has passed an LDAP query for membership in a group. Or verify that the user has passed Active Directory authentication.
    c) To add claims and scopes, click **Add/Delete**; (this opens a popup screen with Scope and Claim tabs); on one or both tabs, select entries and click **Update** (this closes the popup screen).
12. Click **Save**.
    The properties screen closes and the policy displays.
13. Add any additional access policy items you require to complete the access policy.

    ***Note:** On the branch of the access policy with OAuth Authorization, do not also assign connectivity resources (as you can with various resource assign access policy items). Doing so causes a validation error on the Allow ending.*

14. Change the ending from **Deny** to **Allow** on any access policy branch on which you want to grant access.
15. Click the **Apply Access Policy** link to apply and activate the changes to the policy.

Creating a client SSL profile for certificate inspection

You configure a client SSL profile to request an SSL certificate from an OAuth client application or an OAuth resource server at the start of the session.

***Note:** You must configure a client SSL profile for an OAuth client application or an OAuth resource server that registered with Access Policy Manager® (APM®) with a certificate type of authentication.*

1. On the Main tab, click **Local Traffic** > **Profiles** > **SSL** > **Client**.
   The Client SSL profile list screen opens.
2. Click **Create**.
   The New Client SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.
4. From the **Parent Profile** list, select **clientssl**.
5. Scroll down to the Client Authentication area.
6. Next to Client Authentication, select the **Custom** check box.
   The settings become available.
7. From the **Client Certificate** list, retain the default setting **ignore**.

   Selecting **ignore** prevents traffic from OAuth client applications and OAuth resource servers from being blocked if they are configured to use a secret for authentication or to use no authentication at all. The BIG-IP® system still requests a certificate from any OAuth client application or OAuth resource server that specifies a certificate type of authentication.
8. From the **Trusted Certificate Authorities** list, make a selection.

   The BIG-IP system supplies a default certificate and a `ca-bundle.crt` file that includes all well-known public certificate authority (CA) certificates for client-side processing.
9. Click **Finished**.

To put this client SSL profile into effect, select it in a virtual server that is configured to accept HTTPS traffic.

Creating a virtual server for OAuth authorization server traffic

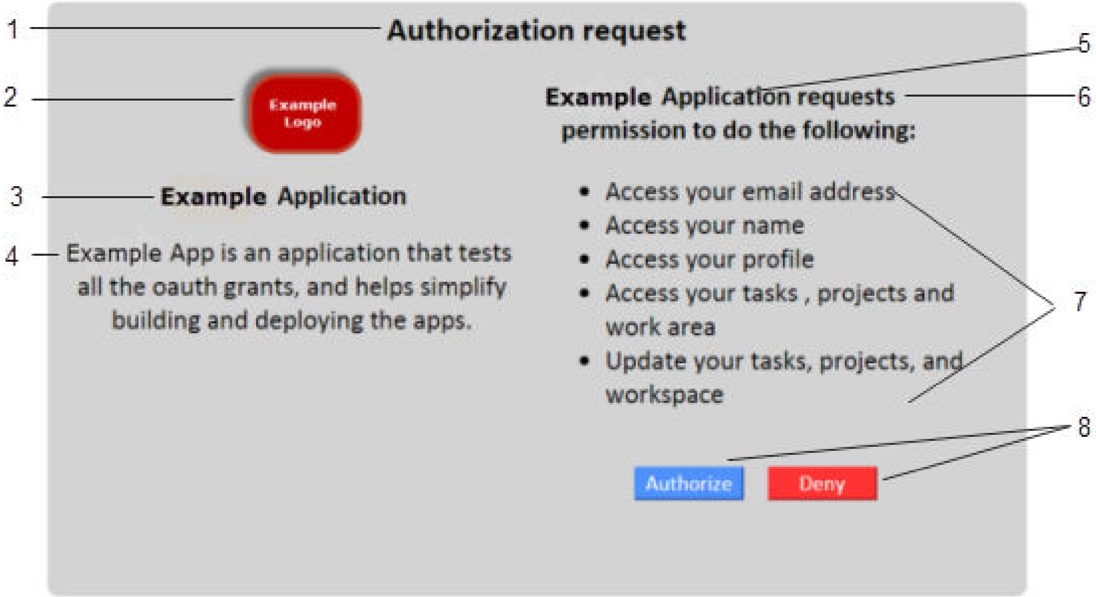
You create a virtual server to process traffic for Access Policy Manager® (APM®) configured as an OAuth authorization server.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Service Port** field, type `443` or select **HTTPS** from the list.
4. From the **HTTP Profile** list, select **http**.
5. For the **SSL Profile (Client)** setting, move the client SSL profile you created earlier to the **Selected** list.
6. Scroll down to the Access Profile area.
7. From the **Access Profile** list, select the access profile you created earlier.
8. Click **Finished**.

The HTTPS virtual server appears in the Virtual Server List screen.

Overview: Localizing an OAuth authorization screen

The text on an OAuth authorization screen is a composite of captions and descriptions configured in a few different objects. When you set out to customize the authorization screen, you need to know where the text comes from.



1
Authorization request
2
Example Logo
3
Example Application
4
Example App is an application that tests all the oauth grants, and helps simplify building and deploying the apps.
5
6
Example Application requests permission to do the following:
- Access your email address
- Access your name
- Access your profile
- Access your tasks , projects and work area
- Update your tasks, projects, and workspace
7
8
Authorize
Deny

Figure 36: An example OAuth Authorization screen

Element	Where configured
1	OAuth Authorization agent, **Authorize Message** field.
2	Client application object, **Website URL Logo** field. (Providing a different logo for different locales is not supported.)
3	Client application object, **Caption** field.
4	Client application object, **Detailed Description** field.
5	Client application object, **Caption** field supplies the application name.
6	OAuth Authorization agent, **Scope Message** field supplies the phrase which defaults to **request permission to do the following**.
7	OAuth scope objects, **Detailed Description** field.
8	OAuth Authorization agent, **Allow Message** and **Deny Message** fields.

Task summary

Localizing an OAuth client application
Localizing an OAuth scope

Localizing an OAuth client application

Before you start, you must have registered a client application in Access Policy Manager® (APM®). When you configure a client application, you specify a caption as you want it displayed when the English language is being used.

You use this process to specify the caption and detailed description as you want them displayed for additional languages.

Note: *APM supports a subset of languages, and, during a session, only matches the languages that are specified in the access profile that started the session.*

Note: *This task covers general customization practices only. For information about advanced customization, see BIG-IP Access Policy Manager: Customization on the AskF5™ web site located at* `support.f5.com`.

---

1. On the Main tab, click **Access** > **Profiles / Policies** > **Customization** > **General**.
   The Customization tool appears in General Customization view, displaying **Form Factor: Full/ Mobile Browser** settings.
2. In the left pane, select the Text tab.
   A navigation tree displays in the left pane.
3. Expand the **OAuth Client Applications** folder.
4. Click the name of the OAuth client application you want to customize.

   The partition precedes the client application name, for example **/Common/myClientApp**.

   A table displays **Name** (setting name) and **Value** (display text) in the right pane.
5. From the **Language** list above the table, select a language.
6. To supply text or to replace existing text for **Caption**:
   a) Click in the **Value** field.
   A pencil icon displays at the end of the field.
   b) Type your translated text and press Enter.
7. To supply text or to replace existing text for **Detailed Description**:
   a) Click in the **Value** field.
   A pencil icon displays at the end of the field.
   b) Type your translated text and press Enter.
8. Click the **Save** icon.
9. To customize text for additional languages, select a language, make the changes that you require, and click **Save** before you select another language.

Localizing an OAuth scope

You must already have a scope configured in Access Policy Manager® (APM®). When you configure a scope, you specify a caption as you want it displayed when the English language is being used.

You use this process to customize the caption and detailed description as you want them displayed for additional languages.

Note: *APM supports a subset of languages, and, during a session, only matches the languages that are specified in the access profile that started the session.*

Note: *This task covers general customization practices only. For information about advanced customization, see BIG-IP Access Policy Manager: Customization on the on the AskF5™ web site located at* `support.f5.com`.

---

1. On the Main tab, click **Access** > **Profiles / Policies** > **Customization** > **General**.
   The Customization tool appears in General Customization view, displaying **Form Factor: Full/ Mobile Browser** settings.
2. In the left pane, select the Text tab.

A navigation tree displays in the left pane.

3. Expand the **OAuth Scopes** folder.
4. Click the name of the OAuth scope that you want to localize.

   The partition where the scope resides precedes the scope name, for example `Common/myEmailScope`.

   A table displays **Name** (setting name) and **Value** (display text) in the right pane.
5. From the **Language** list above the table, select a language.
6. To supply text or to replace existing text for **Caption**:
   a) Click in the **Value** field.
   A pencil icon displays at the end of the field.
   b) Type your translated text and press Enter.
7. To supply text or to replace existing text for **Detailed Description**:
   a) Click in the **Value** field.
   A pencil icon displays at the end of the field.
   b) Type your translated text and press Enter.
8. Click the **Save** icon.
9. To customize text for additional languages, select a language, make the changes that you require, and click **Save** before you select another language.

About customization for policy agents

If an access or per-request policy agent supports customization, customization settings are available in agent properties from within the visual policy editor. The same customization settings are also available for the agent in the **Access** > **Profiles / Policies** > **Customization** area of the BIG-IP® system. For more information, see *BIG-IP® Access Policy Manager® (APM®) Customization* on the AskF5™ web site located at `support.f5.com`.

Overview: Managing opaque access tokens

Access Policy Manager® (APM®) stores access tokens in on-disk databases for their lifetimes.

Task summary

Revoking opaque access tokens
Purging opaque tokens from a database instance
Obtaining a list of OAuth IDs for purged access tokens

Revoking opaque access tokens

To perform this task, you must be logged into the BIG-IP® system in one of these user roles: manager, administrator, or resource administrator.

Based on your observations, you might want to revoke one or more access tokens.

Note: *If the BIG-IP system is part of a high availability pair, you should revoke an access token from the active device. If you revoke an access token from the standby device, the access token retains its ACTIVE status on the active device.*

1. On the Main tab, click **Access** > **Overview** > **OAuth Reports** > **Tokens**.
   The OAuth Tokens screen opens. By default, the `/Common/oauthdb/` default database instance is selected. The display initially places the most recently issued access tokens at the top of the table.

2. Locate the access tokens that you want to delete using any or all of these methods:
    a) From the **DB Instance** list, select the database instance where the client app or the resource server stores tokens.

       If you have few database instances, you might search them in turn for the access tokens that you want to revoke.

       The OAuth profiles that are associated with client apps and resource servers specify which database instance to use.
    b) In the **Search** field, type all or part of a user name or client application name and press Enter.

       Any results that match display.
    c) Sort the tokens by clicking a table heading, such as **Client App** or **Access Token Issued**.
3. Select the access tokens that you want to revoke.
4. Click **Revoke**.
   A popup screen displays a confirmation message.
5. On the popup screen, click **Revoke**.

Purging opaque tokens from a database instance

You purge OAuth database instances of expired tokens when you want to recover disk space.

Note: *Schedule a database purge at a time when the BIG-IP® system is least-used, to prevent any possible performance issues.*

1. On the Main tab, click **Access** > **Federation** > **OAuth Authorization Server** > **Database Instance**.
   The Database Instance screen opens.
2. In the **Name** field, click the name of a database instance.
   A Properties screen opens.
3. To purge the database immediately, below the Purge Schedule Settings area, click **Purge Now**.
4. To specify a schedule for regular database purging, perform these steps:
    a) Select a frequency from the **Frequency** list.
    b) Specify a time in the **Schedule At** field.

    a) Click **Update**.

Obtaining a list of OAuth IDs for purged access tokens

If you want to see a list of OAuth IDs that identify the opaque access tokens that have been purged, you can do so from the command line on the BIG-IP® system.

1. Log on to the command line of the BIG-IP system.
2. Open the TMOS Shell (`tmsh`).
   `tmsh`
3. Type this command: `show apm oauth purged-entries`.
   A list of OAuth IDs displays with the dates and times on which access tokens were purged.

```
    oauth_id                                                purged_time
---------------------------------------------------------------------------
07c64b01e360f43ff4e2b561107de9f4aa5ca14e54b5e72e        2016-09-29 02:00:01
```

About OAuth statistics collection

Access Policy Manager® (APM®) collects OAuth performance statistics on the BIG-IP® system. After you configure and start to use APM as an OAuth server or an OAuth client and resource server, APM collects statistics without requiring any additional setup.

Charting OAuth server performance

To perform this task, you must be logged into the BIG-IP® system in one of these user roles: manager, administrator, or resource administrator.

You chart server performance when you want to see the rate of requests that the OAuth authorization server, Access Policy Manager® (APM®) , processes over a period of time.

1. On the Main tab, click **Access** > **Overview** > **OAuth Reports** > **Server**.

   The Overview chart summarizes OAuth authorization server activity. Additional charts provide statistics by grant type.

   The x-axis for a chart specifies time. The y-axis specifies the rate: requests per second.

   The screen displays OAuth server performance charts with data from the last hour.
2. To display the rate of requests for the interval represented by a point in a chart, place your cursor at the point in the chart.
   The legend updates to display the rates.
3. To get the number of requests for a point in the chart, multiply the rate by the number of seconds in the interval.

   The interval between two consecutive points on a chart depends on the selected **Time**.
4. To generate charts for another time period, select one from the **Time** list.

OAuth performance chart intervals

The interval between two consecutive points in an OAuth performance chart depends on the time period selected for the chart.

Time	Interval
Last hour	30 seconds
Last 3 hours	1 minute (60 seconds)
Last 12 hours	6 minutes (360 seconds)
Last day	12 minutes (720 seconds)
Last week	1 hour (3600 seconds)
Last 30 days	4 hours (14400 seconds)
Last 3 months	12 hours (43200 seconds)
Last 6 months	1 day (86400 seconds)

Charting OAuth opaque token usage

To perform this task, you must be logged into the BIG-IP® system in one of these user roles: manager, administrator, or resource administrator.

You can report on the opaque access tokens that are stored on the OAuth authorization server. You might want to view access and refresh token usage for particular users, client apps, user agents, scopes, or other token data over a time period.

***Note:** To conserve database space and make room for new access tokens, database instances must be purged of expired tokens on a regular basis.*

1. On the Main tab, click **Access** > **Overview** > **OAuth Reports** > **Tokens**.
   The OAuth Tokens screen opens. By default, the `/Common/oauthdb/` default database instance is selected. The display initially places the most recently issued access tokens at the top of the table.
2. Use the fields at the top of the screen (**DB Instance**, **Access Token Issued**, and **Search**) to select a database instance, to change the time period, or to type part of a user or client application name and search for it.
   The screen updates with data to match the filters you set.
3. To view the scopes associated with an access token:
   a) Click the link in the **Scopes** column.
      A popup screen displays the list of scopes.
   b) Click **OK**.
4. To view the user agents associated with an access token:
   a) Click the link in the **User Agents** column.
      A popup screen displays the list of scopes.
   b) Click **OK**.
5. To sort the access tokens by **Client App**, **User**, **User Agent**, or any other table column, click the table column name.
6. To page through the report, use the fields at the bottom of the screen.

Opaque access token status

The OAuth access token report displays a status for each opaque access token. This table defines each status.

Access Token Status	Description
`ACTIVE`	A token status is active when the token is granted and remains active until an event occurs that changes the status.
`EXPIRED`	Token status changes to expired only when a validation request is attempted on a token that has passed its expiration date.
`REVOKED`	Token status changes to revoked when a client or an administrator revokes that access token.

OAuth authorization server troubleshooting tips

You might run into problems with an OAuth authorization server on the BIG-IP® system in some instances. Follow these tips to try to resolve any issues you might encounter.

Log message	Possible explanations and corrective actions
`Invalid grant type requested in OAuth mode`	An OAuth profile might not be specified in the access profile. Verify that the access profile specifies an OAuth profile.
`OAuth mode not set for Authorization`	Incoming OAuth request might not match the configured OAuth endpoints.

Log message	Possible explanations and corrective actions
Agent: Incoming OAuth request might not match the configured OAuth endpoints or could be failing for other reasons.	
OAuth mode not set for Authorization Agent: OAuth profile is not configured for this access profile.	OAuth profile is not specified in the access profile.
The client app does not support Auth code grant	The Authorization Code grant type is not selected in the client app configuration.
The client app does not support Implicit grant	The Implicit grant type is not selected in the client app configuration.
The client app does not support ROPC grant	The Resource Owner Password Credentials grant type is not selected in the client app configuration.
OAuth mode not set for Authorization Agent	An OAuth profile might not be specified in the access profile. Verify that the access profile specifies an OAuth profile.
Invalid Scope: '*name*'	The client application sent a request with an invalid scope, *name*. • The scope might not be configured on the BIG-IP system. • The scope might not be assigned to the client application. Verify and correct the configuration on the BIG-IP system: **1.** Look for the scope. If it does not exist, create it. **2.** Look at the client application. If the scope is not assigned to it, assign it.
Client ID *0fb9b2...* IP 165.160.15.20 attempted to use Auth Code *03f59e...* given to client ID *093eb2...*	A client application tried to use an authorization code that the Authorization Server provided to another client application. Any remediation action, such as unregistering the app, is at the admin's discretion.
Client ID *093eb2...* IP 165.160.15.20 attempted to use already consumed Auth Code *03f59e...*	An authorization code can be used to retrieve an access token once only. This error message indicates that the Authorization server detected a client application presenting the same authorization code to retrieve an access token more than once. Any remediation action, such as notifying the app developer or unregistering the app, is at the admin's discretion.
Failed to initiate DB synchronization (ERR_DB)	The error code might help indicate what problem was seen. This error can also occur if the OAuth plugin restarted.

Log message	Possible explanations and corrective actions
Request Introspect Token from ID *bd3d27...* IP 165.160.15.20 failed. Error Code (invalid_request) Error Description (Required parameter (resource_server_secret) is missing)	The error description field provides the detailed reason why a request failed. The reason could vary from missing a required field in the request to an out-of-memory situation in the traffic management microkernel (TMM) process, and so on. The error description should be detailed enough to help with troubleshooting.
Request Auth Code from Source ID *052ae66...* IP 165.160.15.20 failed. Error Code (server_error) Error Description (Assigned scopes exceed buffer size limit.)	All assigned scopes (space separated) are returned to the client application. However, if all assigned scope names exceed 1000 bytes, this error message will be generated. To resolve the problem, you can: • Shorten the scope names, or • Reduce the number of assigned scopes, or • Both of the above. ***Note:*** *The maximum length for one scope name is 400 characters. The maximum length for all the scope names assigned to the client app and separated by spaces is 1000 characters (1000 bytes).*
Request Auth Code from Source ID *052ae6...* IP 165.160.15.20 failed. Error Code (server_error) Error Description (Assigned scopes cause scope_data to exceed buffer size limit.)	Scope data (scope_data) is JSON-formatted output that contains all assigned scope names and their values. Whenever the JSON-formatted output length exceed 4000 bytes, this error is generated. To resolve the problem, you can: • Shorten the scope name, or • Shorten the scope value, or • Reduce the number of assigned scopes, or • All of the above ***Note:*** *The maximum length for one scope name is 400 characters. The maximum length for one scope value is 3500 characters. The maximum length for all scope data (scope names, scope values, spaces, and formatting characters) is 4000 characters (4000 bytes).*
Failed to register OAuth global tmstat table (ERR_MEM) Failed to create OAuth global tmstat row (ERR_MEM) Failed to set OAuth global tmstat field name (ERR_MEM)	These are OAuth global TMSTAT initialization related failures. These events are unlikely. Restarting TMM could help. For more information, refer to SOL89999342: BIG-IP daemons (12.x) on the AskF5™ web site located at support.f5.com.
Failed to get OAuth global stats row during tmstat initialization (ERR_UNKNOWN)	This is an OAuth global TMSTAT initialization related failure. This is an unlikely event. Restarting TMM could help.

Log message	Possible explanations and corrective actions
`Request Access Token from Source ID 052ae666629882d29c0e385ce9380023e96cf9c0a5ae4857 IP 10.192.144.45 failed.Error Code (server_error) Error Description (JWT AT signing failed)`	Indicates that the JSON web token (JWT) access token signature generation failed. It might fail due to an invalid JSON web key (JWK) configuration being used on the OAuth Profile.
`Request Access Token from Source ID 052ae666629882d29c0e385ce9380023e96cf9c0a5ae4857 IP 10.192.144.45 failed. Error Code (server_error) Error Description (JWT AT signing failed due to expired cert)`	Indicates that the certificate used by the assigned JWK on the OAuth Profile has expired. Either create a configuration with valid certificate or enable the **Ignore expired certificate validation** check box on the OAuth Profile.
`Request Access Token from Source ID 052ae666629882d29c0e385ce9380023e96cf9c0a5ae4857 IP 10.192.144.45 failed. Error Code (server_error) Error Description (Unexpected: JWT subject JSON format out of bound.)`	Indicates the subject field value length is too large. Change the configuration to reduce the size of subject's value.
`Request Access Token from Source ID 052ae666629882d29c0e385ce9380023e96cf9c0a5ae4857 IP 10.192.144.45 failed.Error Code (server_error) Error Description (Generate JWT Access/Refresh token size exceeds`	This indicates the generated access token size is too large than the currently supported size limit of 16 K. Create a configuration using fewer claims and scopes, shorten the claim and scope values, and so on.

Log message	Possible explanations and corrective actions
`available buffer size limit.`	
`Request Access Token from Source ID 052ae666629882d29c0e385ce9380023e96cf9c0a5ae4857 IP 10.192.144.45 failed. Error Code (server_error) Error Description (Refresh token encryption failed)`	While generating the JWT refresh token, encryption failed. This indicates a problem with the **JWT Refresh Token Encryption Secret** configuration or an issue with crypto operations.

Logging and Reporting

Overview: Configuring remote high-speed APM and SWG event logging

You can configure the BIG-IP® system to log information about Access Policy Manager® (APM®) and Secure Web Gateway events and send the log messages to remote high-speed log servers.

When configuring remote high-speed logging of events, it is helpful to understand the objects you need to create and why, as described here:

Object	Reason
Pool of remote log servers	Create a pool of remote log servers to which the BIG-IP system can send log messages.
Destination (unformatted)	Create a log destination of Remote High-Speed Log type that specifies a pool of remote log servers.
Destination (formatted)	If your remote log servers are the ArcSight, Splunk, or Remote Syslog type, create an additional log destination to format the logs in the required format and forward the logs to a remote high-speed log destination.
Publisher	Create a log publisher to send logs to a set of specified log destinations.
Log Setting	Add event logging for the APM system and configure log levels for it or add logging for URL filter events, or both. Settings include the specification of up to two log publishers: one for access system logging and one for URL request logging.
Access profile	Add log settings to the access profile. The log settings for the access profile control logging for the traffic that comes through the virtual server to which the access profile is assigned.

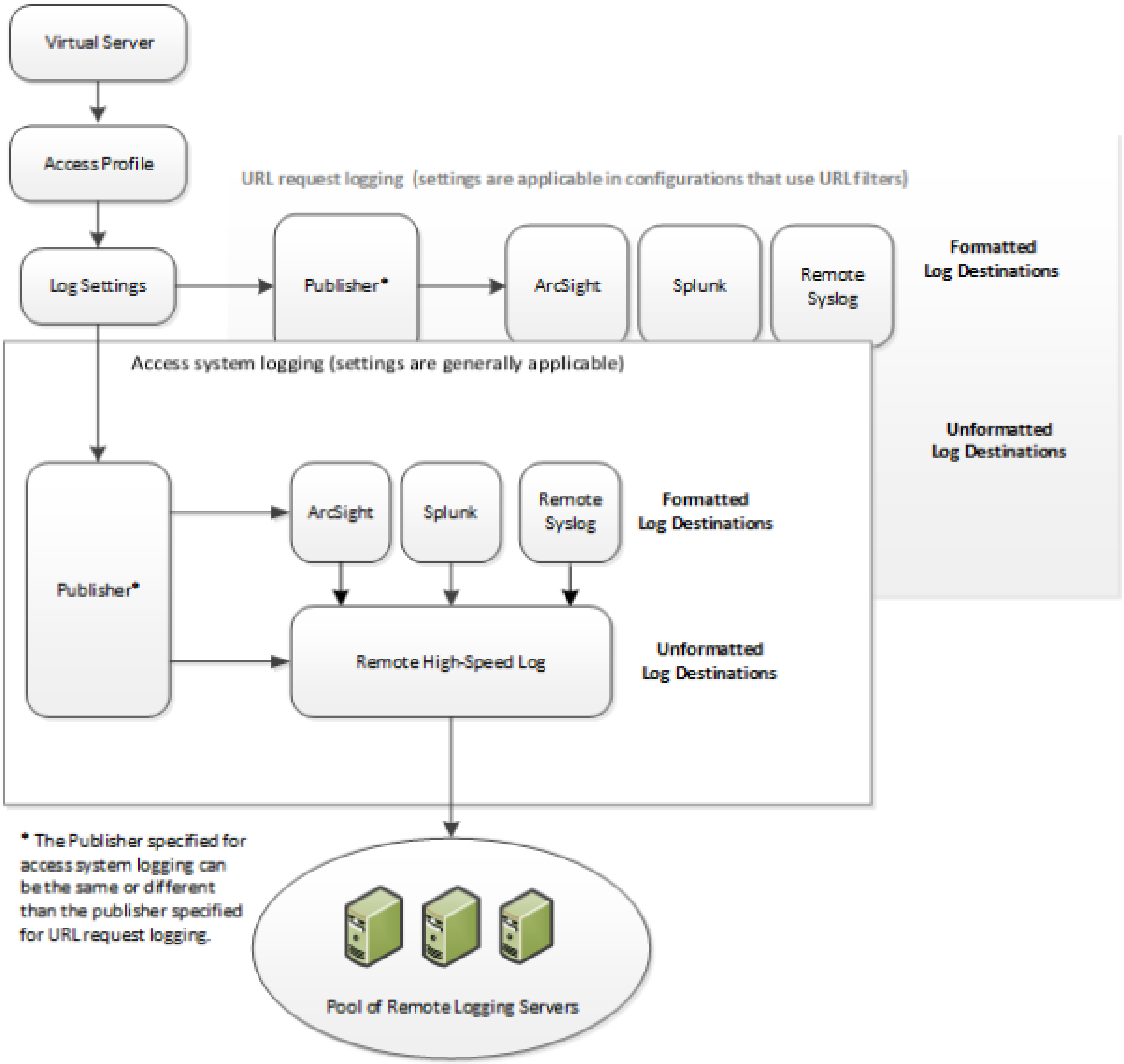

Virtual Server
Access Profile
Log Settings
URL request logging (settings are applicable in configurations that use URL filters)
Publisher*
ArcSight
Splunk
Remote Syslog
Formatted Log Destinations
Unformatted Log Destinations
Access system logging (settings are generally applicable)
Publisher*
ArcSight
Splunk
Remote Syslog
Formatted Log Destinations
Remote High-Speed Log
Unformatted Log Destinations
* The Publisher specified for access system logging can be the same or different than the publisher specified for URL request logging.
Pool of Remote Logging Servers

Figure 37: Association of remote high-speed logging configuration objects

Task summary

Perform these tasks to configure remote high-speed APM and SWG event logging on the BIG-IP system.

Note: *Enabling remote high-speed logging impacts BIG-IP system performance.*

Task list

Creating a pool of remote logging servers
Creating a remote high-speed log destination
Creating a formatted remote high-speed log destination
Creating a publisher
Configuring log settings for access system and URL request events
Disabling logging

About the default-log-setting

Access Policy Manager® (APM®) provides a default-log-setting. When you create an access profile, the default-log-setting is automatically assigned to it. The default-log-setting can be retained, removed, or replaced for the access profile. The default-log-setting is applied to user sessions only when it is assigned to an access profile.

Regardless of whether it is assigned to an access profile, the default-log-setting applies to APM processes that run outside of a user session. Specifically, on a BIG-IP® system with an SWG subscription, the default-log-setting applies to URL database updates.

Creating a pool of remote logging servers

Before creating a pool of log servers, gather the IP addresses of the servers that you want to include in the pool. Ensure that the remote log servers are configured to listen to and receive log messages from the BIG-IP® system.

Create a pool of remote log servers to which the BIG-IP system can send log messages.

1. On the Main tab, click **Local Traffic** > **Pools**.
   The Pool List screen opens.
2. Click **Create**.
   The New Pool screen opens.
3. In the **Name** field, type a unique name for the pool.
4. Using the **New Members** setting, add the IP address for each remote logging server that you want to include in the pool:
   a) Type an IP address in the **Address** field, or select a node address from the **Node List**.
   b) Type a service number in the **Service Port** field, or select a service name from the list.

   Note: *Typical remote logging servers require port* `514`.

   c) Click **Add**.
5. Click **Finished**.

Creating a remote high-speed log destination

Before creating a remote high-speed log destination, ensure that at least one pool of remote log servers exists on the BIG-IP® system.

Create a log destination of the **Remote High-Speed Log** type to specify that log messages are sent to a pool of remote log servers.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Destinations**.
   The Log Destinations screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this destination.
4. From the **Type** list, select **Remote High-Speed Log**.

   Important: *If you use log servers such as Remote Syslog, Splunk, or ArcSight, which require data be sent to the servers in a specific format, you must create an additional log destination of the required type, and associate it with a log destination of the* ***Remote High-Speed Log*** *type. With this configuration, the BIG-IP system can send data to the servers in the required format.*

   The BIG-IP system is configured to send an unformatted string of text to the log servers.

5. From the **Pool Name** list, select the pool of remote log servers to which you want the BIG-IP system to send log messages.
6. From the **Protocol** list, select the protocol used by the high-speed logging pool members.
7. Click **Finished**.

Creating a formatted remote high-speed log destination

Ensure that at least one remote high-speed log destination exists on the BIG-IP® system.

Create a formatted logging destination to specify that log messages are sent to a pool of remote log servers, such as Remote Syslog, Splunk, or ArcSight servers.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Destinations**.
   The Log Destinations screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this destination.
4. From the **Type** list, select a formatted logging destination, such as **Remote Syslog**, **Splunk**, or **ArcSight**.

   The Splunk format is a predefined format of key value pairs.

   The BIG-IP system is configured to send a formatted string of text to the log servers.
5. If you selected **Remote Syslog**, then from the **Syslog Format** list select a format for the logs, and then from the **High-Speed Log Destination** list, select the destination that points to a pool of remote Syslog servers to which you want the BIG-IP system to send log messages.

   Important: *For logs coming from Access Policy Manager® (APM®), only the BSD Syslog format is supported.*
6. If you selected **Splunk**, then from the **Forward To** list, select the destination that points to a pool of high-speed log servers to which you want the BIG-IP system to send log messages.

   The Splunk format is a predefined format of key value pairs.
7. Click **Finished**.

Creating a publisher

Ensure that at least one destination associated with a pool of remote log servers exists on the BIG-IP® system.

Create a publisher to specify where the BIG-IP system sends log messages for specific resources.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this publisher.
4. For the **Destinations** setting, select a destination from the **Available** list, and click **<<** to move the destination to the **Selected** list.

   Note: *If you are using a formatted destination, select the destination that matches your log servers, such as Remote Syslog, Splunk, or ArcSight.*
5. Click **Finished**.

Configuring log settings for access system and URL request events

Create log settings to enable event logging for access system events or URL filtering events or both. Log settings specify how to process event logs for the traffic that passes through a virtual server with a particular access profile.

1. On the Main tab, click **Access** > **Overview** > **Event Logs** > **Settings**.
   A log settings table screen opens.
2. Select a log setting and click **Edit** or click **Create** for a new APM® log setting.
   A popup screen opens with General Information selected in the left pane.
3. For a new log setting, in the **Name** field, type a name.
4. To specify logging, select one or both of these check box options:

   - **Enable access system logs** - This setting is generally applicable. It applies to access policies, per-request policies, Secure Web Gateway processes, and so on. When you select this check box, **Access System Logs** becomes available in the left pane.
   - **Enable URL request logs** - This setting is applicable for logging URL requests when you have set up a BIG-IP® system configuration to categorize and filter URLs. When you select this check box, **URL Request Logs** becomes available in the left pane.

   Important: *When you clear either of these check boxes and save your change, you are not only disabling that type of logging, but any changes you made to the settings are also removed.*

5. To configure settings for access system logging, select **Access System Logs** from the left pane.
   Access System Logs settings display in the right panel.
6. For access system logging, from the **Log Publisher** list select the log publisher of your choice.

   A log publisher specifies one or more logging destinations.

   Important: *The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*

7. For access system logging, retain the default minimum log level, **Notice**, for each option.

   You can change the minimum log level, but **Notice** is recommended.

Option	Description
Access Policy	Events that occur while an access policy runs.
Per-Request Policy	Events that occur while a per-request policy runs.
ACL	Events that occur while applying APM access control lists.
SSO	Events that occur during single-sign on.
Secure Web Gateway	Events that occur during URL categorization on a BIG-IP® system with an SWG subscription.
ECA	Events that occur during NTLM authentication for Microsoft Exchange clients.
OAuth	Events that occur while APM, as an OAuth authorization server, processes requests.
PingAccess Profile	Events related to PingAccess authentication. ***Important:*** *For PingAccess authentication, only the log levels defined in default-log-settings apply.*

Option	Description
VDI	Events related to connections to virtual desktop resources.
Endpoint Management System	Events related to connections to an endpoint management system.

8. To configure settings for URL request logging, select **URl Request Logs** from the left pane. URL Request Settings settings display in the right panel.
9. For URL request logging, from the **Log Publisher** list, select the log publisher of your choice.

   A log publisher specifies one or more logging destinations.

   Important: *The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*
10. To log URL requests, you must select at least one check box option:
    - **Log Allowed Events** - When selected, user requests for allowed URLs are logged.
    - **Log Blocked Events** - When selected, user requests for blocked URLs are logged.
    - **Log Confirmed Events** - When selected, user requests for confirmed URLs are logged.

    Whether a URL is allowed, blocked, or confirmed depends on both the URL category into which it falls, and the URL filter that is applied to the request in the per-request policy.
11. (Optional) To assign this log setting to multiple access profiles now, perform these substeps:

    Note: *Up to three log settings for access system logs can be assigned to an access profile. If you assign multiple log settings to an access profile, and this results in duplicate log destinations, logs are also duplicated.*

    a) Select **Access Profiles** from the left pane.
    b) Move access profiles between the **Available** and the **Selected** lists.

    Note: *You can delete (and add) log settings for an access profile on the Logs page for the access profile.*

    Note: *You can configure the log destinations for a log publisher from the Logs page in the System area of the product.*
12. Click **OK**.
    The popup screen closes. The table displays.

To put a log setting into effect, you must assign it to an access profile. Additionally, the access profile must be assigned to a virtual server.

Disabling logging

Disable event logging when you need to suspend logging for a period of time or you no longer want the BIG-IP® system to log specific events.

Note: *Logging is enabled by adding log settings to the access profile.*

1. To clear log settings from access profiles, on the Main tab, click **Access** > **Profiles / Policies**.
2. Click the name of the access profile.
   Access profile properties display.
3. On the menu bar, click **Logs**.
4. Move log settings from the **Selected** list to the **Available** list.

5. Click **Update**.

Logging is disabled for the access profile.

About event log levels

Event log levels are incremental, ranging from most severe (**Emergency**) to least severe (**Debug**). Setting an event log level to **Warning** for example, causes logging to occur for warning events, in addition to events for more severe log levels. The possible log levels, in order from highest to lowest severity are:

- **Emergency**
- **Alert**
- **Critical**
- **Error**
- **Warning**
- **Notice** (the default log level)
- **Informational**
- **Debug**

***Note:** Logging at the **Debug** level can increase the load on the BIG-IP® system.*

APM log example

The table breaks a typical Access Policy Manager® (APM®) log entry into its component parts.

An example APM log entry

```
Feb  2 12:37:05 site1 notice tmm[26843]: 01490500:5: /Common/for_reports:Common: bab0ff52:
New session from
client IP 10.0.0.1 (ST=/CC=/C=) at VIP 20.0.0.1 Listener /Common/site1_http
(Reputation=Unknown)
```

Information Type	Example Value	Description
Timestamp	**Feb 2 12:37:05**	The time and date that the system logged the event message.
Host name	**site1**	The host name of the system that logged the event message. Because this is typically the host name of the local machine, the appearance of a remote host name could be of interest.
Log level	**notice**	The text value of the log level for the message.
Service	**tmm**	The process that generated the event.
PID	**[26843]**	The process ID.
Log ID	**01490500**	A code that signifies the product, a subset of the product, and a message number.
Level	**5**	The numeric value of the log level for the message.

Information Type	Example Value	Description
Partition	**/Common/for_reports:Common**	The partition.to which configuration objects belong.
Session ID	**bab0ff52**	The ID associated with the user session.
Log message	**New session from client IP 10.0.0.1 (ST=/CC=/C=) at VIP 20.0.0.1 Listener /Common/ site1_http (Reputation=Unknown)**	The generated message text.

About local log destinations and publishers

The BIG-IP® system provides two local logging destinations:

local-db

Causes the system to store log messages in the local MySQL database. Log messages published to this destination can be displayed in the BIG-IP Configuration utility.

local-syslog

Causes the system to store log messages in the local Syslog database. Log messages published to this destination are not available for display in the BIG-IP Configuration utility.

***Note:** Users cannot define additional local logging destinations.*

The BIG-IP system provides a default log publisher for local logging, sys-db-access-publisher; initially, it is configured to publish to the local-db destination and the local-syslog destination. Users can create other log publishers for local logging.

Configuring a log publisher to support local reports

APM® provides preconfigured reports that are based on log data. To view the reports and to display log data from the BIG-IP® Configuration utility, configure a publisher to log to the local-db destination.

***Important:** The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Select the log publisher you want to update and click **Edit**.
3. For the **Destinations** setting, select **local-db** from the **Available** list, and move the destination to the **Selected** list.
4. Click **Finished**.

To use a log publisher, specify it in an access policy log setting, ensure that the access profile selects the log setting, and assign the access profile to a virtual server.

***Note:** Log settings are configured in the **Access** > **Overview** > **Event Log** > **Settings** area of the product.*

Viewing an APM report

If Access Policy Manager® (APM®) events are written to the local database on the BIG-IP® system, they can be viewed in APM reports.

Create a report to view event log data.

1. On the Main tab, click **Access** > **Overview** > **Access Reports**.

   The Reports Browser displays in the right pane. The Report Parameters popup screen opens and displays a description of the current default report and default time settings.
2. (Optional) Select the appropriate **Restrict by Time** settings.
3. Click **Run Report**.
   The popup screen closes. The report displays in the Reports Browser.

You can select and run various system-provided reports, change the default report, and create custom reports.

Viewing URL request logs

To view URL request logs from the user interface, your access profile log setting must enable URL request logs. The log setting must also specify a log publisher that publishes to the local-db log destination.

You can display, search, and export URL request logs.

1. On the Main tab, click **Access** > **Overview** > **Event Logs** > **URL Request Logs**.

   Any logs for the last hour are displayed.

   Note: *APM® writes logs for blocked requests, confirmed requests, allowed requests, or all three, depending on selections in the access profile log setting.*
2. To view logs for another time period, select it from the list.
3. To search the logs, type into the field and click **Search** or click **Custom Search** to open a screen where you can specify multiple search criteria.
4. To export the logs for the time period and filters, click **Export to CSV**.

Configuring a log publisher to supply local syslogs

If you must have syslog files available on the local device, configure a publisher to log to the local-syslog destination.

Important: *The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Select the log publisher you want to update and click **Edit**.
3. For the **Destinations** setting, select **local-syslog** from the **Available** list, and move the destination to the **Selected** list.
4. Click **Finished**.

To use a log publisher, specify it in an access policy log setting, ensure that the access profile selects the log setting, and assign the access profile to a virtual server.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product.*

Preventing logging to the /var/log/apm file

To stop logs from being written to the /var/log/apm file, remove the local-syslog destination from log publishers that are specified for access system logging in APM® log settings.

Important: *The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Select the log publisher you want to update and click **Edit**.
3. For the **Destinations** setting, if the **Selected** list contains **local-syslog**, move it to the **Available** list.
4. Click **Finished**.

To use a log publisher, specify it in an APM log setting, ensure that the log setting is assigned to an access profile, and assign the access profile to a virtual server.

Note: *Log settings are configured in the* ***Access*** *>* ***Overview*** *>* ***Event Log*** *>* ***Settings*** *area of the product.*

About local log storage locations

The BIG-IP® system publishes logs for portal access traffic and for connections to virtual desktops (VDI) to the `/var/log/rewrite*` files. APM® cannot publish these logs to remote destinations.

APM can publish URL request logs to remote or local destinations. Logs published to the local-db destination are stored in the local database and are available for display from the Configuration utility. Logs published to the local-syslog destination are stored in the `/var/log/urlfilter.log` file.

APM can publish access system logs to remote or local destinations. Logs published to the local-db destination are stored in the local database. Logs in the local database are available for display in APM reports. Logs published to the local-syslog destination are stored in the `/var/log/apm` file.

Code expansion in Syslog log messages

The BIG-IP® system log messages contain codes that provide information about the system. You can run the Linux command `cat` *`log`* `|bigcodes |less` at the command prompt to expand the codes in log messages to provide more information. For example:

```
   Jun 14 14:28:03 sccp bcm56xxd [ 226 ] : 012c0012 : (Product=BIGIP Subset=BCM565XXD) :
6: 4.1 rx [ OK 171009 Bad 0 ] tx [ OK 171014 Bad 0 ]
```

About configurations that produce duplicate log messages

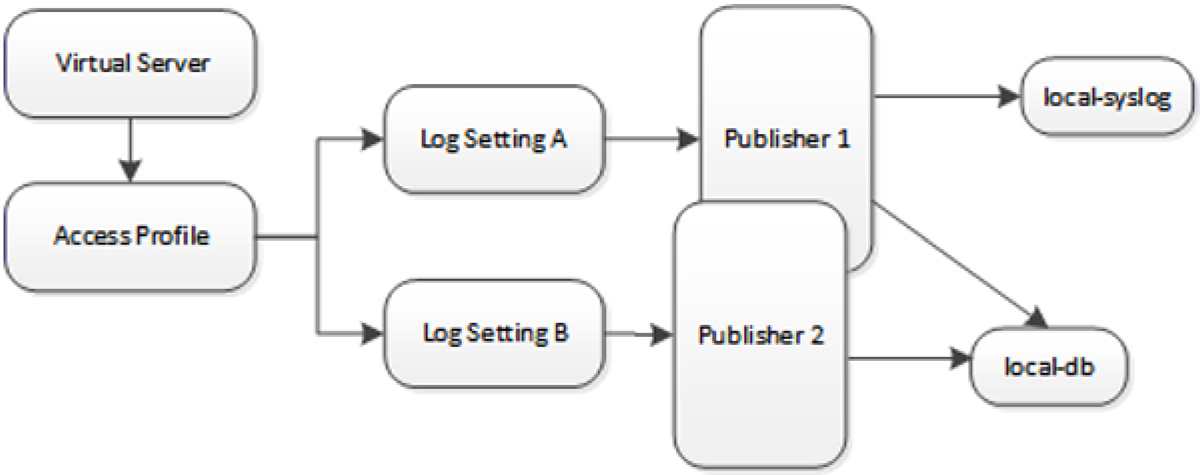


Figure 38: Event log duplication

The figure illustrates a configuration that writes duplicate logs. Two log publishers specify the same log destination, local-db. Each log publisher is specified in one of the log settings that are assigned to an access profile. Logs are written to the local-db destination twice.

Methods to prevent or eliminate duplicate log messages

Duplicate log messages are written when the same log destination is specified by two or more log publishers and more than one of the log publishers is specified in the log settings that are assigned to an access profile.

One way to avoid or eliminate this problem is to specify only one log setting for each access profile. Another is to ensure that the log publishers you associate with log settings for an access profile do not contain duplicate log destinations.

About log level configuration

Log levels can be configured in various ways that depend on the specific functionality. Log levels for access portal traffic are configured in the System area of the product. The log level for the URL database download is configured in the default-log-setting in the **Access** > **Overview** > **Event Log** > **Settings** area of the product. The log level for NTLM authentication of Microsoft Exchange clients is configured using the ECA option in any log setting. Other access policy (and Secure Web Gateway) log levels are configured in any log setting.

Updating the log level for NTLM for Exchange clients

Before you follow these steps, you must have an access profile that you configured to use for NTLM authentication of Microsoft Exchange clients. You must know the name of the log setting that is assigned to that access profile. (The default-log-setting is assigned by default, but your access profile configuration might be different.)

You can change the level of logging for NTLM authentication for Microsoft Exchange clients.

*Note: Logging at the default level, **Notice**, is recommended.*

1. On the Main tab, click **Access** > **Overview** > **Event Logs** > **Settings**.
   A log settings table screen opens.
2. Select the check box for the log setting that you want to update and click **Edit**.
   A popup screen opens.
3. To configure settings for access system logging, select **Access System Logs** from the left pane.
   Access System Logs settings display in the right panel.
4. For the **ECA** setting, select a log level.

   *Note: Setting the log level to **Debug** can adversely impact system performance.*
5. Click **OK**.
   The popup screen closes.

Configuring logging for the URL database

Configure logging for the URL database so that log messages are published to the destinations, and at the minimum log level, that you specify. (Logging for the URL database occurs at the system level, not the session level, and is controlled using the default-log-setting log setting.)

Note: A URL database is available only on a BIG-IP® system with an SWG subscription.

1. On the Main tab, click **Access** > **Overview** > **Event Logs** > **Settings**.
   A log settings table screen opens.
2. From the table, select **default-log-setting** and click **Edit**.
   A log settings popup screen displays.
3. Verify that the **Enable access system logs** check box is selected.
4. To configure settings for access system logging, select **Access System Logs** from the left pane.
   Access System Logs settings display in the right panel.
5. From the **Log Publisher** list, select the log publisher of your choice.

   A log publisher specifies one or more logging destinations.

   ***Important:** The BIG-IP® system is not a logging server and has limited capacity for storing, archiving, and analyzing logs. For this reason a dedicated logging server is recommended.*
6. To change the minimum log level, from the **Secure Web Gateway** list, select a log level.

   *Note: Setting the log level to **Debug** can adversely impact system performance.*

   The default log level is **Notice**. At this level, logging occurs for messages of severity Notice and for messages at all incrementally greater levels of severity.
7. Click **OK**.
   The popup screen closes. The table displays.

Setting log levels for Portal Access events

Change the logging level for access policy events when you need to increase or decrease the minimum severity level at which Access Policy Manager® (APM®) logs that type of event. Follow these steps to change the log level for events that are related to portal access traffic.

Note: You can configure log levels for additional APM options in the Event Logs area.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Options**.
2. Scroll down to the Access Policy Logging area.

   Note: *The log settings that you change on this page impact only the access policy events that are logged locally on the BIG-IP® system.*

3. For **Portal Access**, select a logging level from the list.

   Warning: *F5® recommends that you do not set the log level for* ***Portal Access*** *to* ***Debug****. Portal Access can stop working. The BIG-IP system can become slow and unresponsive.*

4. Click **Update**.

APM starts to log events at the new minimum severity level.

Legal Notices

Legal notices

Publication Date

This document was published on October 22, 2018.

Publication Number

MAN-0506-06

Copyright

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

For a current list of F5 trademarks and service marks, see *http://www.f5.com/about/guidelines-policies/trademarks*.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by one or more patents indicated at: *https://f5.com/about-us/policies/patents*.

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

RF Interference Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This unit generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This Class A digital apparatus complies with Canadian ICES-003.

Standards Compliance

This product conforms to the IEC, European Union, ANSI/UL and Canadian CSA standards applicable to Information Technology products at the time of manufacture.

Index